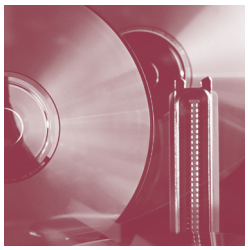
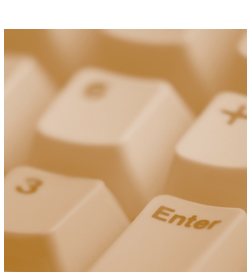
ASG-Manager Products™ Installation in CMS Environments

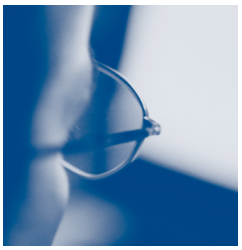
Version: 2.5

Publication Number: MPR0300-25-INCMS

Publication Date: November 1996







ASG Worldwide Headquarters Naples, Florida USA | asg.com
1333 Third Avenue South, Naples, Florida 34102 USA Tel: 941.435.2200 Fax: 941.263.3692 Toll Free: 1.800.932.5536

ASG Documentation/Product Enhancement Fax Form

Please FAX comments regarding ASG products and/or documentation to (941) 263-3692.

Company Name	Telephone Number	Site ID	Contact name

Product Name/Publication	Version #	Publication Date
Product:		
Publication:		
Tape VOLSER:		

Enhancement Request:

ASG Support Numbers

ASG provides support throughout the world to resolve questions or problems regarding installation, operation, or use of our products. We provide all levels of support during normal business hours and emergency support during non-business hours. To expedite response time, please follow these procedures.

Please have this information ready:

- Product name, version number, and release number
- List of any fixes currently applied
- Any alphanumeric error codes or messages written precisely or displayed
- A description of the specific steps that immediately preceded the problem
- The severity code (ASG Support uses an escalated severity system to prioritize service to our clients. The severity codes and their meanings are listed below.)

If You Receive a Voice Mail Message:

1. Follow the instructions to report a production-down or critical problem.
2. Leave a detailed message including your name and phone number. A Support representative will be paged and will return your call as soon as possible.
3. Please have the information described above ready for when you are contacted by the Support representative.

Severity Codes and Expected Support Response Times

Severity	Meaning	Expected Support Response Time
1	Production down, critical situation	Within 30 minutes
2	Major component of product disabled	Within 2 hours
3	Problem with the product, but customer has work-around solution	Within 4 hours
4	"How-to" questions and enhancement requests	Within 4 hours

ASG provides software products that run in a number of third-party vendor environments. Support for all non-ASG products is the responsibility of the respective vendor. In the event a vendor discontinues support for a hardware and/or software product, ASG cannot be held responsible for problems arising from the use of that unsupported version.

Business Hours Support

Your Location	Phone	Fax	E-mail
United States and Canada	800.354.3578 1.941.435.2201 **Secondary Numbers:** 800.227.7774 800.525.7775	941.263.2883	support@asg.com
Australia	61.2.9460.0411	61.2.9460.0280	support.au@asg.com
England	44.1727.736305	44.1727.812018	support.uk@asg.com
France	33.141.028590	33.141.028589	support.fr@asg.com
Germany	49.89.45716.300	49.89.45716.400	support.de@asg.com
Singapore	65.224.3080	65.224.8516	support.sg@asg.com
All other countries:	1.941.435.2201		support@asg.com

Non-Business Hours - Emergency Support

Your Location	Phone	Your Location	Phone
United States and Canada	800.354.3578 1.941.435.2201 **Secondary Numbers:** 800.227.7774 800.525.7775 **Fax:** 941.263.2883		
Asia	011.65.224.3080	**Japan/Telecom**	0041.800.9932.5536
Australia	0011.800.9932.5536	**New Zealand**	00.800.9932.5536
Denmark	00.800.9932.5536	**South Korea**	001.800.9932.5536
France	00.800.9932.5536	**Sweden/Telia**	009.800.9932.5536
Germany	00.800.9932.5536	**Switzerland**	00.800.9932.5536
Hong Kong	001.800.9932.5536	**Thailand**	001.800.9932.5536
Ireland	00.800.9932.5536	**United Kingdom**	00.800.9932.5536
Israel/Bezeq	014.800.9932.5536		
Japan/IDC	0061.800.9932.5536	**All other countries**	1.941.435.2201

ASG Web Site

Visit http://www.asg.com, ASG's World Wide Web site.

Submit all product and documentation suggestions to ASG's product management team at http://www.asg.com/products/suggestions.asp

If you do not have access to the web, FAX your suggestions to product management at (941) 263-3692. Please include your name, company, work phone, e-mail ID, and the name of the ASG product you are using. For documentation suggestions include the publication number located on the publication's front cover.

Contents

Preface

This *ASG-Manager Products Installation in CMS Environments* provides information about installing ASG-Manager Products (herein called Manager Products) in a CMS environment.

Within the Manager Products family, ASG-ControlManager (herein called ControlManager) and ASG-DictionaryManager (herein called DictionaryManager) are co-requisites of each other. Both are environmental prerequisites (EPR) and must be at the latest version and release level for each and every other Manager Product to execute correctly.

Allen Systems Group, Inc. (ASG) provides professional support to resolve any questions or concerns regarding the installation or use of any ASG product. Telephone technical support is available around the world, 24 hours a day, 7 days a week.

ASG welcomes your comments, as a preferred or prospective customer, on this publication or on any ASG product.

About this Publication

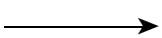
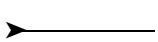
This publication consists of these chapters:

- Chapter 1, "Overview of Installation," summarizes the features of Manager Products, including installation, tailoring, and dictionary features.
- Chapter 2, "Installing an Initial Manager Products Environment,"describes how an initial installation can be set up in a CMS environment. This chapter also describes how to create and load an MP-AID, create and restore a Demo Dictionary, and run Manager Products.
- Chapter 3, "Step 1 - Copy Datasets to Disk,"explains how to copy datasets to disk.
- Chapter 4, "Step 2 - Select and Generate an Executable Version of Manager Products,"describes the three options for executing Manager Products. The options include CMS LOADLIB mode, discontiguous saved segment (DCSS) mode, and GENMOD mode.
- Chapter 5, "Step 3 - Tailoring Installation Macros and Modules Supplied in MP.CMSRCE,"contains table summarizations displaying contents of the dataset MP.SOURCE. This chapter also describes installation macros, source modules, DesignManager FORMAT members, and DictionaryManager COMMAND-STREAM members.

- Chapter 6, "Step 4 - Create and Load the MP-AID,"describes how to create a BDAM-Organized MP-AID, load dataset MP.INFO.UNLOAD and load dataset MP.COM.UNLOAD.
- Chapter 7, "Step 5 - Set Up Dictionaries,"explains how to create a Manager Products BDAM Dictionary, Administration Dictionary, Demo Dictionary, and user interface (POST) Dictionary.
- Chapter 8, "Step 6 - Satisfy Concurrent Usage Requirements,"contains procedures for establishing an environment where concurrent access can occur without the risk of corruption and which is transparent to all users.
- Chapter 9, "Step 7 - Linking External Software To Manager Products Software," provides details of how to link external software to Manager Products software.
- Chapter 10, "Step 8 - Run Manager Products with BDAM MP-AID and BDAM Dictionaries,"outlines the installation steps for running Manager Products using a BDAM MP-AID and Dictionary.
- Chapter 11, "Introduction to Installing MethodManager,"summarizes the features of MethodManager and references appropriate sections for information regarding installation.
- Chapter 12, "Installing An Initial MethodManager Environment,"describes how an initial MethodManager installation can be set up in a CMS environment.
- Chapter 13, "Installing an Operational MethodManager Environment,"contains the steps required to perform an operational ASG-MethodManager installation in a CMS environment.
- Chapter 14, "Setting up an MP-AID and/or Repositories from an Existing Manager Products Installation,"outlines the steps necessary to set up a separate ASG-MethodManager installation which uses repositories and an MP-AID from an existing Manager Products installation.

Publication Conventions

Diagrams are read from left to right along a continuous line (the "main path"). Keywords and variables appear on, above, or below the main path.

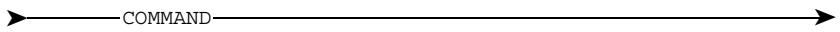
Convention	Represents
	At the beginning of a line indicates the start of a statement.
	At the end of a line indicates the end of a statement.
	At the end of a line indicates that the statement continues on the line below.
	At the beginning of a line indicates that the statement continues from the line above.

Keywords are in upper-case characters. Keywords and any required punctuation characters or symbols are highlighted. Permitted truncations are not indicated.

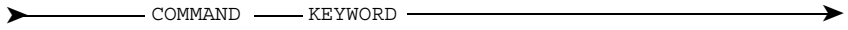
Convention	Represents

Variables are in lower-case characters.

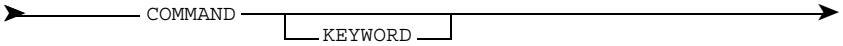
Statement identifiers appear on the main path of the diagram:



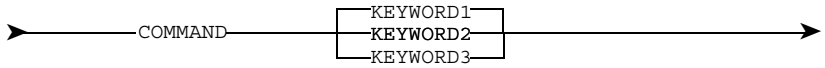
A required keyword appears on the main path:



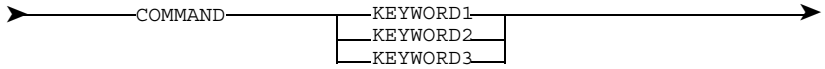
An optional keyword appears below the main path:



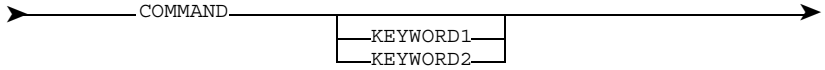
Where there is a choice of required keywords, the keywords appear in a vertical list; one of them is on the main path:



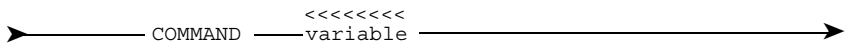
or



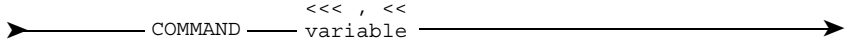
Where there is a choice of optional keywords, the keywords appear in a vertical list, below the main path:



The repeat symbol, <<<<<<, above a keyword or variable, or above a whole clause, indicates that the keyword, variable, or clause may be specified more than once:

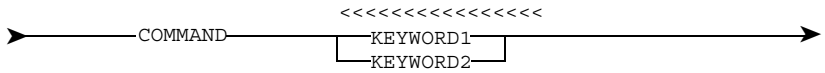


A repeat symbol broken by a comma indicates that if the keyword, variable, or clause is specified more than once, a comma must separate each instance of the keyword, variable, or clause:

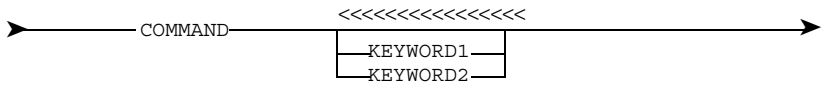


Convention	Represents

The repeat symbol above a list of keywords (one of which appears on the main path) indicates that any one or more of the keywords may be specified; at least one must be specified:



The repeat symbol above a list of keywords (all of which are below the main path) indicates that any one or more of the keywords maybe specified, but they are all optional:



Allen Systems Group, Inc. uses these conventions in technical publications:

Convention	Represents
ALL CAPITALS	Directory, path, file, dataset, member, database, program, command, and parameter names.
Initial Capitals on Each Word	Window, field, field group, check box, button, panel (or screen), option names, and names of keys. A plus sign (+) is inserted for key combinations (e.g., Alt+Tab).
lowercase italic monospace	Information that you provide according to your particular situation. For example, you would replace *filename* with the actual name of the file.
Monospace	Characters you must type exactly as they are shown. Code, JCL, file listings, or command/statement syntax. Also used for denoting brief examples in a paragraph.
Vertical Separator Bar (|) with underline	Options available with the default value underlined (e.g., Y|N).

1 Overview of Installation

The Manager Family

Manager Products all have the same basic method of installation. ASG supplies you with a magnetic tape or cartridge which contains all the Manager Products software that you have purchased. This software contains your Manager Products nuclei and your optional additional facilities (for example, Manager Products integrators).

ControlManager provides, via its Integrated Dialog Directory, the gateway for users to access the capabilities of the installed Manager Products. Thus, ControlManager is the end user facility for Manager Products and is a prerequisite for the use of any Manager Products.

If you are installing MethodManager, you should note that Chapter 1, "Overview of Installation," on page 11, to Chapter 10, "Step 8 - Run Manager Products with BDAM MP-AID and BDAM Dictionaries," on page 101 are generally applicable. Any difference or additional installation requirements for MethodManager are described in Chapter 11, "Introduction to Installing MethodManager," on page 111 through Chapter 14, "Setting up an MP-AID and/or Repositories from an Existing Manager Products Installation," on page 127.

Before undertaking any installation tasks you are advised to print out the contents of dataset MP.README.

This dataset contains a section describing any revisions that have taken place to the installation process subsequent to the publication of the current installation manuals.

You can execute these CMS commands to print this dataset from the release tape (assuming the release tape is available at virtual device 181):

```
TAPE REW
FILEDEF INMOVE TAP1 SL
FILEDEF OUTMOVE PRINTER
MOVEFILE
```

The Implementation of Manager Products Under VM/CMS

Manager Products executes in VMISP, VM/XA, or VM/ESA environments using the facilities provided by the Conversational Monitor System (CMS) component of VM. In VM/XA or VM/ESA environments Manager Products software will execute in either 370 or XA/ESA mode.

You can use the CP SET MACHINE 370/XA/ESA command to define the virtual machine execution mode prior to starting a Manager Products session.

When Manager Products functionality is executed under CMS, the software that is invoked is essentially that which is used when running an OS version (for example MVS/TSO). The default simulation mode provided by CMS is OS and it is in this mode that Manager Products executes, irrespective of the operating system specified when the Manager Products software is ordered (for example, DOS/VSE). You do not need to use the CMS SET DOS ON command to establish a CMS/DOS environment.

When running under CMS, Manager Products are designed to be executed interactively in full screen mode. However, they are also capable of being executed in CMS batch, running in a disconnected virtual machine.

There are a number of points which may be of interest to DOS users executing under CMS:

- The facilities provided by the DataManager IMS (DL/I) Definition (selectable unit DMR-DD5) and the DataManager IMS (DL/I) Generation (selectable unit DMR-5L5) are as described in the publication *ASG-DataManager DL/I Interface: DOS*, not the publication *ASG-DataManager IMS (DL/I) Interface*.
- If you have a DataManager Source Language Generation selectable unit, generated output from the PRODUCE command can only be directed to a sequential file. The default control cards generated depend on the operating system specified by the user when ordering Manager Products:
  - — For VSE the control cards output are CATALS and BKEND.
  - — For VSE/SP the control cards output are CATALOG and /+.
- If you use a DataManager Automation of Set Up facility (selectable unit DMR-ASU) or DMR-AS2), you should be aware of this information:
  - — With non-CMS execution, input to a CONVERT command must be from a sequential file and must contain the appropriate BKEND statements as described in the publication *ASG-DataManager Automation Of Set-up*.
  - — With CMS execution, input to a CONVERT command must be from a partitioned dataset (that is, a CMS file with a file type of MACLIB) and must not contain any BKEND statements.

Setting Up a Manager Products Installation

Introduction

The flexibility provided by Manager Products together with their optional additional facilities ensures that the installation requirements of your organization can be achieved. In order to gain the optimum value from your Manager Products software, ASG recommends that you identify and define these categories of user:

The systems administrator. The systems administrator is responsible for setting up the environment in which all users will work in a Manager Products installation. The systems administrator is responsible for the overall security and smooth running of the installation, in particular for the MP-AID (Manager Products Administration and Information Dataset). See the *ASG-Manager Products System Administrator's Guide* for full details of the systems administrator's role.

The dictionary controllers. A dictionary controller is responsible for the administration and security of a particular Manager Products dictionary. The capabilities directly available to a controller are effective only in a particular dictionary, whereas the systems administrator has capabilities available that are effective throughout an installation. See the *ASG-Manager Products Controller's Manual* for full details of the controller's role.

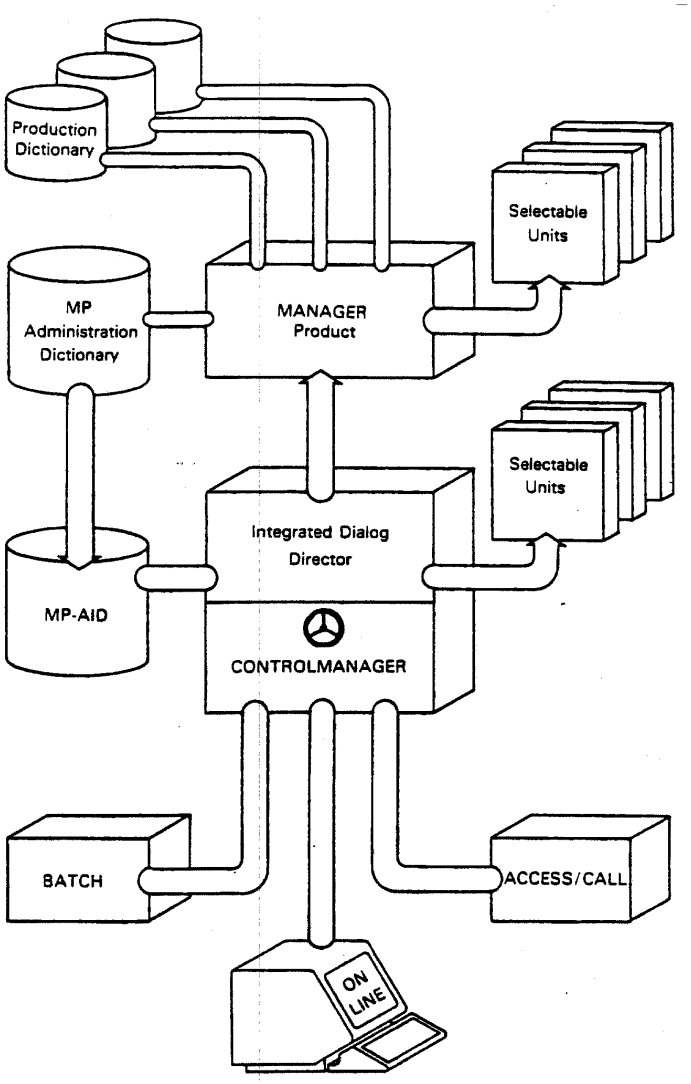
The general user. It is important that both the dictionary controllers and the systems administrator identify who the users are to be and the needs of those users. The environment in which a particular user will work can then be set by using both the controller's and systems administrator's capabilities.

The Architecture of a Manager Products Installation

Once you have identified your systems administrator, dictionary controllers, and users, the next step is to establish how you can optimally utilize the capabilities available with your Manager Products software. The capabilities available will depend on the Manager Products and optional additional facilities purchased.

Figure 1 shows the architecture of a Manager Products installation in which ControlManager and one other Manager Products are installed. This figure forms the background for the following sections, which discuss the steps to be taken when installing your Manager Products software and the factors to be considered at each step.

Figure 1. Basic Architecture of a Manager Products Installation



Installing from the ASG-supplied Tape

This manual contains separate instructions for setting up an initial Manager Products installation and setting up an operational Manager Products installation. Most job control statements given in this manual are also supplied as EXECs in dataset MP.CMS. The EXEC name corresponding to a given figure in this publication is indicated in an initial comment line in each case.

The first step for either type of installation is to copy certain datasets to disk. In the case of an initial installation you need only copy dataset MP.CMS (which contains all the executable versions of Manager Products software) to disk. For an operational installation it will probably be necessary to copy some additional supplied datasets to disk. For a list of datasets which may be supplied on your Manager Products Release tape(s), please refer to Appendix C,"Attributes of Datasets Which May Be Supplied on Your Release Tape," on page 149.

If you have purchased several Manager Products, together with the necessary Manager Products Integrators, the versions of the Manager Products installed will be automatically integrated and no further action need be taken in this regard.

It is necessary to run Manager Products in line mode when creating an MP-AID. Thereafter, Manager Products are available for interactive use in full screen mode.

Tailoring Manager Products Software

ASG supplies tailoring macros, which enable certain options to be fixed on installation. The tailoring of your Manager Products by use of these macros is a tailoring of the installed version of Manager Products (not a provision of runtime options). Thus, it provides one method by which the systems administrator can set up the environment for all users. Installation macros are described in Chapter 5, "Step 3 - Tailoring Installation Macros and Modules Supplied in MP.CMSRCE," on page 37.

The Manager Products Administration and Information Dataset (MP-AID)

The MP-AID is the dataset that contains the information about your Manager Products and your dictionaries in addition to the information used to control the environment in which your Manager Products will operate. The systems administrator uses the MP-AD to define environmental controls that can be applied to all users, particular groups of users, or individual users.

Thus, once you have installed your Manager Products software and performed any tailoring using the installation macros, it is recommended that you create the dataset which when loaded will become your MP-AID (see Chapter 6, "Step 4 - Create and Load the MP-AID," on page 75). When this dataset has been created, the systems administrator will be able to log on to ControlManager.

The systems administrator’s first task should be to load onto the MP-AID:

- The ASG-supplied InfoBank (which provides online documentation of your Manager Products software)
- The ASG-supplied COMMAND members, corporate executive routines, and user defined syntax tables.

If you have purchased the Corporate Dictionary/Repository Definition Export to IDD (selectable unit DYR-TE08), ASG recommends that you load the ASG-supplied translation rules.

Your installation’s own environmental controls are first defined in a dictionary and then loaded onto the MP-AID by your systems administrator. ASG recommends that a separate dictionary, the Manager Products Administration Dictionary, be set up to hold the definitions that are to be loaded onto the MP-AID.

Manager Products Dictionaries

The Dictionary Datasets

A Manager Products dictionary comprises four or five datasets (files):

- Source
- Data Entries
- Index
- Recovery
- Log (if logging is applied to the dictionary)

The Source dataset contains member definitions as input or as subsequently amended.

The Data Entries dataset contains encoded definitions generated by the dictionary management software from the definitions in the Source dataset. It contains information about a member's relationships with other members.

The Index dataset contains the name of each member in the dictionary and is designed to achieve the fastest possible retrieval of source or encoded definitions.

The Recovery dataset is used as a temporary backup file for the dictionary's automatic recovery system.

The Log dataset records all updating commands (or all commands issued, if so specified), together with any associated member definitions or amendments input. It also records full date, time, user, status, and the number of physical I/Os that have occurred.

The Manager Products Administration Dictionary

ASG recommends that your installation set up a Manager Products Administration Dictionary (see "Set Up Manager Products Administration Dictionary" on page 84). It is intended that this dictionary will contain all the data required by ControlManager and will be maintained by the systems administrator. The systems administrator will then have a secure dictionary in which to develop the member definitions which are to be loaded onto the MP-AID (see "The Manager Products Administration and Information Dataset (MP-AID)" on page 15).

If you have purchased one or more of the following optional additional facilities, ASG supplies a dictionary for your use and reference:

- With the ControlManager User Defined InfoSystem facility (selectable unit CMR-UD10), ASG's InfoDictionary is supplied. (See *ASG-Manager Products User Defined InfoSystem* for full details of this dictionary.)
- With the ControlManager User Defined Syntax facility (selectable unit CMR-UD1), ASG's UDS Table Dictionary is supplied. (See the *ASG-Manager Products Controller's Manual* for full details of this dictionary.)
- With the DictionaryManager Corporate Dictionary/Repository Definition Export For IDD facility (selectable unit DYR-TE08), a set of translation rules is supplied.

ASG recommends that, where applicable, these ASG-supplied dictionaries be restored into your Manager Products Administration Dictionary.

If you have any of the following selectable units installed, the ASG-supplied executive routines can be restored into your Manager Products Administration Dictionary for subsequent tailoring and/or maintenance:

- Corporate Dictionary/Repository Definition Import from DB2 (selectable unit DYR-TI12)
- Corporate Dictionary/Repository Definition Export to DB2 (selectable unit DYR-TE12)
- Corporate Dictionary/Repository Definition Import from SQL/DS (selectable unit DYR-TI32)
- Corporate Dictionary/Repository Definition Export to SQL/DS (selectable unit DYR-TE32)
- Workstation Interface (selectable unit CMR-WS0l). The executive routines are needed only if you wish to use the Repository Diagram Generation feature.
- ADW/IEW Integration Facility (selectable units TE14, TE15, TI14, and TI15)

The DEMO Dictionary

The DEMO Dictionary is the ASG-supplied dictionary that is designed for use with the *ASG-DataManager Example Book*. The dictionary demonstrates the features available with all your Manager Products. Therefore, ASG recommends that you install this dictionary while you are setting up your installation so that user training can begin immediately.

The DEMO Dictionary can be retrieved and set up as described in the *ASG-Manager Products Controller's Manual*. The particular example job control requirements are given in this publication in "Set Up the Manager Products DEMO Dictionary" on page 87.

User Interface (POST) Dictionary

If you have purchased the User Interface Facility (selectable unit CMR-UI1) and/or the User Defined Output facility (selectable unit DYR-UD15), a dataset MP.UIDICT is included on the installation tape. This dataset contains a dictionary of the User Interface output record formats and Access Call control parameter area definitions, and information relating to User Defined Output parameter numbers. The POST dictionary can be retrieved and set up as described in the *ASG-Manager Products Controller's Manual*. The particular example job control statements are given in this publication in "Set Up the User Interface (POST) Dictionary" on page 88.

Production Dictionaries

Your Manager Products are designed to work on and be driven by dictionaries. Thus, the next step in setting up your installation is to create the dictionaries that you require. The job control requirements for dictionary creation are given in "Creating Dictionaries" on page 82. The Manager Products commands required for dictionary creation are discussed in the *ASG-Manager Products Controller's Manual*.

Preparing to Run the Generic Import Example

In order to execute the Bachman generic import example you need to perform these installation steps:

1. Copy the dataset that forms the input to the run from the release tape to a suitable disk. The required dataset is named MP.SAMPLE.IMPORT and can be copied to disk using the CMS MOVEFILE command.

2. Make available the Bachman example executive routines. These are provided in status BACHMAN on the Manager Products DEMO dictionary, supplied as dataset MP.DEMO. These executive routines must be encoded and then constructed onto an appropriate MP-AID.

3. Update your Manager Products execution EXEC to define the input dataset as created above. The required ddname for this dataset is DD1.

Tailoring the ADW/IEW Integration Facility

Before you can successfully execute the ADW/IEW Integration facility there are some tailoring tasks that must be performed.

This mandatory tailoring is common to all operating environments and for this reason is documented in the *ASG-Manager Products Tools Support: Integration with ADW/IEW*.

2 Installing an Initial Manager Products Environment

Introduction

This chapter describes how an initial installation can be set up (that is, without considering the effect of selectable units or installation macros) in a CMS environment. If you are a new user, this initial installation allows you to experiment and gain a degree of familiarity with Manager Products before setting up an operational Manager Products installation (described in Chapter 3, "Step 1 - Copy Datasets to Disk," on page 27 through Chapter 10, "Step 8 - Run Manager Products with BDAM MP-AID and BDAM Dictionaries," on page 101). It consists of these components:

- An executable version of your Manager Products software
- The Manager Products Administration and Information Dataset (MP-AID)
- The ASG-supplied InfoBank (loaded onto the MP-AID)
- The ASG-supplied COMMAND members (loaded onto the MP-AID)
- The ASG-supplied Corporate Executive Routines (loaded onto the MP-AID)
- The Manager Products DEMO Dictionary.

The setting up of an initial installation is described in this chapter in four steps:

Step 1. Install an executable version of Manager Products

Step 2. Create and load an MP-AID

Step 3. Create and restore a DEMO dictionary

Step 4. Run Manager Products.

After following these steps ASG recommends that you use InfoBank and the DEMO Dictionary interactively to develop your knowledge of Manager Products and their use.

These assumptions have been made in this chapter:

- The Manager Products software has been installed on the 191 disk of the CMS logon ID identified here as MPR1
- The MP-AID is to be created on the 191 disk of the CMS logon ID identified here as MPR2 and is formatted with a blocksize of 4,096 bytes
- The DEMO dictionary is to be created on the 191 disk of the CMS logon ID identified here as MPR3 and is formatted with a blocksize of 4,096 bytes
- The MP-AID and the DEMO dictionary, when created, are not configured for access for more than one user. Should you wish to configure either the MP-AID or the DEMO dictionary for concurrent usage, you should follow the instructions given in Chapter 8, "Step 6 - Satisfy Concurrent Usage Requirements," on page 89, which describes in full the steps necessary to achieve such usage. If concurrent usage is not implemented and the MP-AID or DEMO dictionary is inadvertently used concurrently, it is likely that both the MP-AID and DEMO dictionary will be corrupted.
- Installation option 3 has been chosen when installing an executable version of ControlManager.

Example values are given in place of variable identifiers within job control statements and Manager Products commands. Example values within the job control statements are underlined in this chapter.

InfoBank can be entered at the highest level by typing `INFOBANK`.

A hard copy record of the personnel system defined in the DEMO dictionary, which includes suggested examples and demonstrations, is given in the *ASG-DataManager Example Book*. Note that at present the *ASG-DataManager Example Book* refers to the DEMO Dictionary as the Example Dictionary.

Step 1 - Install an Executable Version of Manager Products

The release tape containing the dataset MP.CMS should be mounted on a tape unit which has been attached at virtual address 181 to the appropriate CMS logon (in this case logon ID MPR1). ASG recommends that the CMS disk be formatted with a blocksize of 2,048 bytes. Dataset MP.CMS contains the Manager Products CMS software and requires 7,000 2K blocks of disk storage. However, this is the basic disk storage requirement and a further 1,000 2K blocks of disk storage will be needed in order to generate an executable version of Manager Products software.

The Manager Products software is copied to the CMS mini-disk using these CMS commands:

```
TAPE REW
TAPE FSF n
TAPPOS * TEXT A1 (UPDATE COL1
```

where *n* is 76 in an OS environment or 40 in a DOS environment.

The CMS installation EXECs, supplied by ASG to simplify installation, should now be made available by entering these CMS commands:

```
RENAME MPX0 TEXT A1 MPX0 EXEC A1
MPX0
```

ASG recommends that you copy the example EXECs MPXI1, MPXI2, MPXI3, and MPXI4 to the CMS system disk or to an extension of the CMS system disk where they can be amended as necessary and made available to all users.

There are three installation options for installing Manager Products in a CMS environment. This document assumes that option 3 is to be selected. (Information concerning all options is given in Chapter 4, "Step 2 - Select and Generate an Executable Version of Manager Products," on page 31). Option 3 involves the creation of an executable module in GENMOD mode; that is, using the CMS GENMOD command. To achieve this you should issue this command:

```
MPX3
```

After successful completion of the above commands, the supplied Manager Products software is available for execution.

Step 2 - Create and Load an MP-AID

Step 2A - Create an MP-AID and Load Dataset MP.INFO.UNLOAD

The dataset MP.INFO.UNLOAD on the release tape contains an unloaded MP-AID which holds the ASG-supplied InfoBank. Before ControlManager can be used, an MP-AID must be built by creating an empty MP-AID and then loading the contents of MP.INFO.UNLOAD.

To start, enter this command:

```
DEFINE STORAGE 3M
IPL CMS
```

You should then run a CMS EXEC containing the CMS commands given in Figure 2. Definitions of the variables used can be found in "Step 4A - Create a BDAM-Organized MP-AID and Load Dataset MP.INFO.UNLOAD" on page 76.

Figure 2. Create and Load a Manager Products BDAM MP-AID with Dataset MP.INFO.UNLOAD

```
/* MPXI1 - Create and Load a Manager Products BDAM MP-AID   */
/*                                                          */
'EXECOS'
'GLOBAL LOADLIB MPRLIB'
'CP LINK MPR1 191 200 R'
'CP LINK MPR2 191 201 MW'
'ACCESS 200 F'
'ACCESS 201 G'
'SET LDRTBLS 10'
'FILEDEF MPIN  TERMINAL'
'FILEDEF MPOUT TERMINAL'
'FILEDEF MPRDIAG DISK MPRDIAG LISTING A1'
'FILEDEF MPAID DISK CMR MPAID G6 (XTENT 3000'
'FILEDEF MPAIDR TAP1 SL 6 (BLOCK 9442'
'LOADMOD MPR00'
'START *LINE'
'GLOBAL LOADLIB'
'RELEASE F (DET'
'RELEASE G (DET'
```

You cannot run ControlManager in full screen mode until the MP-AID has been created. To create and load the MP-AID, enter the following Manager Products commands. These commands must be entered in line mode:

```
MP-AID CREATE ADMINISTRATOR SYSAD PASSWORD SAD
PHYSICAL-BLOCKSIZE 8192 LOGICAL-BLOCKSIZE 1024;
LOGON SYSAD PASSWORD SAD;
MP-AID BUFFERS 30;
MP-AID LOAD INFOBANK
MP-AID STATUS;
LOGOFF;
```

The allocation of an increased size MP-AID buffer pool enhances performance. A minimum allocation of 30 buffers is recommended.

Note:

The number of blocks allocated to the MP-AID is based on the number of blocks required to install the ASG-supplied InfoSystem, UDS tables, COMMAND members, and corporate executive routines, together with about 10 percent available for user storage, if required.

Step 2B - Load the MP-AID With Dataset MP.COM.UNLOAD

The dataset MP.COM.UNLOAD on the release tape contains the ASG-supplied COMMAND members, corporate executive routines, and UDS tables needed to execute certain Manager Products functionality.

The installation of this dataset is mandatory.

To load MP.COM.UNLOAD you should first enter these commands:

```
DEFINE STORAGE 3M
IPL CMS
```

You should then run a CMS EXEC containing the CMS commands given in Figure 3. Definitions of the variables used can be found in "Step 4B - Load the MP-AID With Dataset MP.COM.UNLOAD" on page 78.

Figure 3. Loading a BDAM MP-AID with Dataset MP.COM.UNLOAD

```
/* MPXI2 - Load the MP-AID with Dataset MP.COM.UNLOAD   */
/*                                                      */
'EXECOS'
'GLOBAL LOADLIB MPRLIB'
'CP LINK MPR1 191 200 R'
'CP LINK MPR2 191 201 MW'
'ACCESS 200 F'
'ACCESS 201 G'
'SET LDRTBLS 10'
'FILEDEF MPIN  TERMINAL'
'FILEDEF MPOUT TERMINAL'
'FILEDEF MPRDIAG DISK MPRDIAG LISTING A1'
'FILEDEF MPAID DISK CMR MPAID G6 (XTENT 3000'
'FILEDEF MPAIDR TAP1 SL 8 (BLOCK 9442'
'LOADMOD MPR00'
'START *'
'GLOBAL LOADLIB'
'RELEASE F (DET'
'RELEASE G (DET'
```

When ControlManager starts executing you will be presented with a logon panel. Enter `SYSAD` as the logon identifier and `SAD` as the password.

Once you have successfully logged on, enter these Manager Products commands:

```
MP-AID BUFFERS 30;
MP-AID LOAD ALL ;
MP-AID STATUS ;
LOGOFF ;
```

Step 3 - Create and Restore a Demo Dictionary

Your Manager Products release tape contains the dataset MP.DEMO, which holds the DEMO dictionary. This dictionary can be retrieved by running ControlManager in order to create an empty dictionary and to restore the DEMO dictionary.

You should first enter these commands:

```
DEFINE STORAGE 4M
IPL CMS
```

You should then run a CMS EXEC containing the CMS commands given in Figure 4 to create a BDAM dictionary. Definitions of the variables used can be found in "Set Up Manager Products Administration Dictionary" on page 84.

Figure 4. Create and Restore a BDAM DEMO Dictionary

```
/* MPXI3 - Create and Restore a BDAM DEMO Dictionary   */
/*                                                     */
'EXECOS'
'GLOBAL LOADLIB MPRLIB'
'CP LINK MPR1 191 200 R'
'CP LINK MPR2 191 201 MW'
'CP LINK MPR3 191 202 MW'
'ACCESS 200 F'
'ACCESS 201 G'
'ACCESS 202 H'
'SET LDRTBLS 10'
'FILEDEF MPIN  TERMINAL'
'FILEDEF MPOUT TERMINAL'
'FILEDEF MPRDIAG DISK MPRDIAG LISTING A1'
'FILEDEF MPAID DISK CMR MPAID G6 (XTENT 3500'
'FILEDEF DEMO  DISK DEMO INDEX H6 (XTENT 20'
'FILEDEF DEMOD DISK DEMO DATAENT H6 (XTENT 220'
'FILEDEF DEMOS DISK DEMO SOURCE H6 (XTENT 270'
'FILEDEF DEMOE DISK DEMO RECOVER H6 (XTENT 120'
'FILEDEF DEMOJ DISK DEMO LOG H6 (XTENT 50'
'FILEDEF DEMOR TAP1 SL 19 (BLOCK 9442'
'LOADMOD MPR00'
'START *'
'GLOBAL LOADLIB'
'RELEASE F (DET'
'RELEASE G (DET'
'RELEASE H (DET'
```

When ControlManager starts executing you will be presented with a logon panel into which you should enter SYSAD for the logon ID and SAD for the password.

Once you have successfully logged on enter these Manager Products commands:

```
CREATE DEMO MASTER CON
ILB 2046 SLB 314 DLB 360 IPB 8192 SPB 4096 DPB 8192
RPB 4096 LPB 4096 WITH 18 STATUSES AND LOG;
DICTIONARY DEMO;
AUTHORITY CON;
RESTORE ALL;
LOGOFF;
```

Step 4 - Run Manager Products

Enter these commands:

```
DEFINE STORAGE 4M
IPL CMS
```

Then run a CMS EXEC containing the CMS commands given in Figure 5 on page 26. Definitions of the variables used can be found in Chapter 10, "Step 8 - Run Manager Products with BDAM MP-AID and BDAM Dictionaries," on page 101.

Figure 5. Running Manager Products (BDAM MP-AID and Dictionary)

```
/* MPXI4 - Running Manager Products (BDAM MP-AID and Dictionary   */
/*                                                                */
'EXECOS'
'GLOBAL LOADLIB MPRLIB'
'CP LINK MPR1 191 200 R'
'CP LINK MPR2 191 201 MW'
'CP LINK MPR3 191 202 MW'
'ACCESS 200 F'
'ACCESS 201 G'
'ACCESS 202 H'
'SET LDRTBLS 10'
'FILEDEF MPIN  TERMINAL'
'FILEDEF MPOUT TERMINAL'
'FILEDEF MPRDIAG DISK MPRDIAG LISTING A1'
'FILEDEF MPAID DISK CMR MPAID G6 (XTENT 3500'
'FILEDEF DEMO  DISK DEMO INDEX H6 (XTENT 20'
'FILEDEF DEMOD DISK DEMO DATAENT H6 (XTENT 220'
'FILEDEF DEMOS DISK DEMO SOURCE H6 (XTENT 270'
'FILEDEF DEMOE DISK DEMO RECOVER H6 (XTENT 120'
'FILEDEF DEMOJ DISK DEMO LOG H6 (XTENT 50'
'LOADMOD MPR00'
'START   *'
'GLOBAL LOADLIB'
'RELEASE F (DET'
'RELEASE G (DET'
'RELEASE H (DET'
```

3 Step 1 - Copy Datasets to Disk

Together, Chapter 3, "Step 1 - Copy Datasets to Disk," on page 27 through Chapter 10, "Step 8 - Run Manager Products with BDAM MP-AID and BDAM Dictionaries," on page 101 describe installing an operational Manager Products environment. This chapter describes the first step, copying datasets to disk.

Introduction

Your Manager Products software is always supplied on tape. You will receive one or more tapes depending on the products purchased. The datasets supplied on these tapes are dependent upon the Manager Products and selectable units which have been ordered. The attributes and usage of the supplied datasets can be found in Appendix C,"Attributes of Datasets Which May Be Supplied on Your Release Tape," on page 149.

The datasets must be copied to a Manager Products software mini-disk to which access can be made by all users wishing to use Manager Products. ASG recommends that the disk be formatted with a blocksize of 2,048 bytes.

The ASG-supplied datasets described in this chapter must be copied to disk if they are to be used.

Copying Dataset MP.CMS

The MP.CMS dataset is always required when installing Manager Products. To copy MP.CMS to disk, enter these commands:

```
TAPE REW
TAPE FSF n
TAPPDS * TEXT A1 (UPDATE COL1
```

where

n is an integer that can be determined from the formula (3*x*-2),

x is the position of the relevant dataset on the ASG-supplied tape. The position of the dataset is given in the list of datasets provided with your ASG-supplied tape(s).

The basic disk storage requirement for MP.CMS is 7,200 2K blocks. Additional blocks will be needed in order to generate an executable version of Manager Products software. Details of storage requirements for generating an executable version of Manager Products are given in Chapter 4, "Step 2 - Select and Generate an Executable Version of Manager Products," on page 31.

Copying Dataset MP.CMSRCE

The MP.CMSRCE dataset is required for tailoring Manager Products. To copy MP.CMSRCE to disk and create a MACLIB, enter:

```
TAPE REW
TAPE FSF n
TAPEMAC name
```

where:

n is an integer that can be determined from the formula (3*x*-2),

x is the position of the relevant dataset on the ASG-supplied tape. The position of the dataset is given in the list of datasets provided with your ASG-supplied tape(s).

name specifies the name of the MACLIB to be created.

Copying Dataset MP.UIBAL.CMS

The MP.UIBAL.CMS dataset may be required if you have the User Interface facility (CMR-UIl). To copy MP.UIBAL.CMS to disk and create a MACLIB, enter:

```
TAPE REW
TAPE FSF n
TAPPDS * MACLIB Al (UPDATE COLl
```

where:

n is an integer that can be determined from the formula (3*x*-2),

x is the position of the relevant dataset on the ASG-supplied tape. The position of the dataset is given in the list of datasets provided with your ASG-supplied tape(s).

On completion of the TAPPDS command, a CMS file called MPUIBAL MACLIB is generated.

To create individual CMS files from the MACLIB you should use the MOVEFILE command.

Copying Dataset MP.UICOB.CMS

The MP.UICOB.CMS dataset may be required if you have the User Interface facility (CMR-UIl). To copy MP.UICOB.CMS to disk and create a MACLIB, enter:

```
TAPE REW
TAPE FSF n
TAPPDS * MACLIB Al (UPDATE COLl
```

where:

n is an integer that can be determined from the formula (3*x*-2),

x is the position of the relevant dataset on the ASG-supplied tape. The position of the dataset is given in the list of datasets provided with your ASG-supplied tape(s).

On completion of the TAPPDS command, a CMS file called MPUICOB MACLIB is generated.

To create individual CMS files from the MACLIB you should use the MOVEFILE command.

Copying Dataset MP.UIPLI.CMS

The MP.UIPLI.CMS dataset may be required if you have the User Interface facility (CMR-UI1). To copy MP.UIPLI.CMS to disk and create a MACLIB, enter:

```
TAPE REW
TAPE FSF n
TAPPDS * MACLIB Al (UPDATE COLl
```

where:

n is an integer that can be determined from the formula (3*x*-2),

x is the position of the relevant dataset on the ASG-supplied tape. The position of the dataset is given in the list of datasets provided with your ASG-supplied tape(s).

On completion of the TAPPDS command, a CMS file called MPUIPLI MACLIB is generated.

To create individual CMS files from the MACLIB you should use the MOVEFILE command.

Copying Dataset MP.UIMIV.CMS

The MP.UIMIV.CMS dataset may be required if you have the User Interface facility (CMR-UI1). To copy MP.UIMIV.CMS to disk and create a MACLIB, enter:

```
TAPE REW
TAPE FSF n
TAPPDS * MACLIB A1 (UPDATE COL1
```

where:

n is an integer that can be determined from the formula (3*x*-2),

x is the position of the relevant dataset on the ASG-supplied tape. The position of the dataset is given in the list of datasets provided with your ASG-supplied tape(s).

On completion of the TAPPDS command, a CMS file called MPUIMIV MACLIB is generated.

To create individual CMS files from the MACLIB you should use the MOVEFILE command.

4 Step 2 - Select and Generate an Executable Version of Manager Products

Together, Chapter 3, "Step 1 - Copy Datasets to Disk," on page 27 through Chapter 10, "Step 8 - Run Manager Products with BDAM MP-AID and BDAM Dictionaries," on page 101 describe installing an operational Manager Products environment. This chapter describes the second step, generating an executable version of Manager Products.

Introduction

Manager Products can be executed using any of these options:

Option 1. CMS LOADLIB mode

Option 2. Discontiguous Saved Segment (DCSS) mode

Option 3. GENMOD mode

The following sections contain information regarding the selection and generation process for each of these 3 options. Any installation of DesignManager is always to a CMS LOADLIB.

For all options you may also install user interface programs in GENMOD mode if required. For details, refer to Chapter 9, "Step 7 - Linking External Software To Manager Products Software," on page 95.

Selecting the CMS Installation Option Appropriate to Your Environment

When installing Manager Products to run under CMS, a number of factors need to be considered in order to select the most appropriate option for your installation:

- The number of users running Manager Products at any one time
- The frequency of use
- The virtual storage size of the installation standard virtual machine. Can the size be increased if necessary? Is it desirable to do so?
- Whether the extra systems effort involved in installing the Manager Products Shareable Code (MPSC) as a Discontiguous Saved Segment (DCSS) is acceptable.

The ideal Manager Products configuration is achieved by using the smallest possible virtual machine, multi-threading the code if possible, and obtaining a good response time.

For an installation where there are a significant number of users, the second option would be the ideal implementation if the overhead of the additional systems effort involved is acceptable. If this overhead is not acceptable, then a choice between options 1 and 3 must be made.

Use of option 3 is likely to require a larger virtual machine than option 1 as the majority of the program code is pre-loaded. This is in contrast to option l, where the program code is loaded in separate segments on demand and can be overlaid if required by use of the systems administrator's SET SEGMENTS command.

Making The Execution Execs Available For Use

Various execution EXECs are supplied by ASG to simplify the installation procedure. Enter these commands to make them available for use:

```
RENAME MPX0 TEXT Al MPX0 EXEC Al
MPX0
```

You can delete the files DSRTXT TXTLIB and MPRTXT TXTLIB once your chosen installation option is successfully installed. However, you must not delete any of the ASG-supplied text files (that is, those files with file type TEXT), as these are used at execution time.

ASG recommends that you copy the example execution EXECs MPX01, MPX02, MPX03, and MPX04 to the CMS system disk or to an extension of the CMS system disk where they can be amended as necessary and made available to all users.

Generation of the Option 1 Executable Version

If you wish to generate a version of Manager Products in CMS LOADLIB mode, enter this command (1,000 2K blocks of disk storage are required):

```
MPX1
```

The name of the generated CMS LOADLIB is MPRLIB.

If you wish to install ASG-DesignManager, you should enter this (700 2K blocks of disk storage are required):

```
MPX5 <OVERLAY>
```

This command will link-edit ASG-DesignManager to MPRLIB. If you wish to tailor DesignManager using the LBUFI macro (see "DesignManager Installation Macros" on page 66) then you should generate an updated version of LBUFl and issue the following commands to update the DesignManager TXTLIB before linking using the MPX5 command:

```
TXTLIB DEL DSRTXT CP00L1
TXTLIB ADD DSRTXT LBUF1
```

The OVERLAY keyword is optional. It allows you to generate an overlaid version of DesignManager, which reduces the virtual storage requirement for DesignManager from approximately 1300K to 800K.

Generation of the Option 2 Executable Version

The Manager Products Shareable Code (MPSC) consists of these software components:

- ControlManager
- DataManager
- DictionaryManager

The installation of the Manager Products Shareable Code (MPSC) in DCSS mode is governed by factors that are particular to a given installation. For example, the MPSC DCSS must be loaded at an address that is beyond the highest address of any virtual machine using it.

These are the steps necessary to install the MPSC DCSS in a VMISP environment:

1. Add the MPSC entry to the System Name Table File (DMKSNT) using the NAMESYS macro. An example MPSC entry is:

```
CMSS2300    NAMESYS SYSNAME=CMSS2300,       *
            SYSVOL=cccccc,                  *
            SYSCYL=,                        *
            SYSSTRT=(mmm,nnn),              *
            SYSPGCT=448,                    *
            SYSHRSG=(pp-qq),                *
            SYSPGNM=(rrr-sss),              *
            SYSSIZE=1792K,                  *
            VSYSRES=,                       *
            VSYSADR=IGNORE
```

   where:

   cccccc is the name of the relevant disk.

   mmm is the starting cylinder address on SYSVOL.

   nnn is the page address on SYSVOL.

pp is the number of the first segment to be shared by the MPSC.

qq is the number of the final segment to be shared by the MPSC.

rrr is the number of the first page to be saved.

sss is the number of the final page to be saved.

2. Include the updated System Name Table File in the Control Program Nucleus.

3. At an address corresponding to that of page *rrr* of the SYSPGNM operand of the NAMESYS macro, install MPSC in GENMOD mode by entering the following command (1,000 2K blocks of disk storage are required):

   ```
   MPX2 page-address
   ```

   where *page-address* is the hexadecimal address at which MPSC is to be installed.

4. Issue these commands to make the MPSC program available in DCSS mode:

   ```
   LOADM MPR00
   CP SAVESYS CMSS2300
   ```

These are the steps necessary to install the MPSC DCSS in a VM/XA or ESA environment:

1. Define the MPSC DCSS using this command:

   ```
   CP DEFSEG CMSS2300 hexpage1-hexpage2 SR
   ```

   where *hexpage1-hexpage2* is the range of pages to be saved. MPSC consists of 448 (X'1C0') pages.

   hexpage1 is obtained by converting the hexadecimal storage address at which MPSC is to be installed into a hexadecimal page number.

   hexpage2 is obtained by adding X' lBF' to *hexpage1*.

2. Except when reloading DCSS after applying fixes, install MPSC in GENMOD mode at a storage address corresponding to *hexpage1* of the above DEFSEG command, by entering this command (1,000 2K blocks of disk storage are required):

   ```
   MPX2 address
   ```

   where *address* is the hexadecimal storage address at which MPSC is to be installed. MPSC must execute in virtual storage below the 16MB line.

3. Issue these commands to make MPSC available in DCSS mode:

   ```
   LOADM  MPR00
   CP SAVESEG CMSS2300
   ```

Example

Assuming that MPSC is to be loaded at a virtual storage address of X '800000', then the commands required to install MPSC are:

```
CP DEFSEG CMSS2300 800-98F SR
MPX2 800000
LOADM MPR00
CP SAVESEG CMSS2300
```

If you wish to install DesignManager, enter this command (700 2K blocks of disk storage are required):

```
MPX5 <OVERLAY>
```

This command link-edits DesignManager into a CMS LOADLIB called MPRLIB. If you wish to tailor DesignManager using the LBUF1 macro (see "DesignManager Installation Macros" on page 66), then you should generate an updated version of LBUF1 and issue these commands to update the DesignManager TXTLIB before linking using the MPX5 command:

```
TXTLIB DEL DSRTXT CP00L1
TXTLIB ADD DSRTXT LBUF1
```

The OVERLAY keyword is optional. It allows you to generate an overlaid version of DesignManager, which reduces the virtual storage requirement for DesignManager from approximately 1,300K to 800K.

Generating the Option 3 Executable Version

Enter this command (1000 2K blocks of disk storage are required):

```
MPX3
```

If you wish to install DesignManager, you should enter this command (700 2K blocks of disk storage are required):

```
MPX5 <OVERLAY>
```

This command link-edits DesignManager into a CMS LOADLIB called MPRLIB. If you wish to tailor DesignManager using the LBUF1 macro (see "DesignManager Installation Macros" on page 66), then you should generate an updated version of LBUF1 and issue these commands to update the DesignManager TXTLIB before linking using the MPX5 command:

```
TXTLIB DEL DSRTXT CP00L1
TXTLIB ADD DSRTXT LBUF1
```

The OVERLAY keyword is optional. It allows you to generate an overlaid version of DesignManager, which reduces the virtual storage requirement for DesignManager from approximately 1,300K to 800K.

5 Step 3 - Tailoring Installation Macros and Modules Supplied in MP.CMSRCE

Introduction

The dataset MP.CMSRCE is supplied by ASG and contains a number of members used in the installation and configuration of a Manager Products environment. The contents of MP.CMSRCE are summarized in Table 1 on page 37. In most cases further details of the SOURCE members can be found in this chapter; but in some cases further details can be found elsewhere, as it is considered that the SOURCE members are best described within the context of general installation details of the corresponding facilities. The SOURCE members on MP.CMSRCE have been divided in this chapter into these categories:

- Installation macros
- Source modules
- DesignManager FORMAT members
- DictionaryManager COMMAND-STREAM members.

Table 1. Contents of Dataset MP.CMSRCE

Member Name	Description	Selectable Unit/ Product	Generated Load Module
DB2PLT	FORMAT member	DSR-UD30, DSR-PH02	-
DB2PRV	FORMAT member	DSR-UD30, DSR-PH02	-
DB2REP	FORMAT member	DSR-UD30, DSR-PH02	-
DCHAR	Tailoring macro	Manager Products	DM195
DCONV	Tailoring macro	DMR-AS1, DMR-AS2	DFU07
DCUST	Tailoring macro	Manager Products	DMU09
DDS2K	Tailoring macro	DMR-SL9	DYD99
DGADA	Tailoring macro	DMR-SL6	DFU16
DGBAL	Tailoring macro	DMR-SL3, DMR-SL5, DMR-SL6, DMR-SL9	DFU10
DGCOB	Tailoring macro	DMR-SL1, DMR-SL5, DMR-SL6, DMR-SL9, DMR-SL10	DFU11
DGDBD	Tailoring macro	DMR-SL5	DIL88
DGMIV	Tailoring macro	DMR-SL7	DFU15
DGPLI	Tailoring macro	DMR-SL5	DFU12

Table 1. Contents of Dataset MP.CMSRCE

Member Name	Description	Selectable Unit/ Product	Generated Load Module
DGPSB	Tailoring macro	DMR-SL5	DIL89
DGREC	Tailoring macro	DMR-SL1, DMR-SL2, DMR-SL3	DFU14
DGSBAL	Tailoring macro	DMR-SL5	DIL97
DGSCOB	Tailoring macro	DMR-SL5	DIL99
DGSPLI	Tailoring macro	DMR-SL5	DIL98
DGSREC	Tailoring macro	DMR-SL5	DIL96
DGS2K	Tailoring macro	DMR-SL9	DYD11
DGTOT	Tailoring macro	DMR-SL4	DFU13
DLOG	Tailoring macro	Manager Products	DML99
DMEX1	Input User Exit	CMR-UI1	DMEX1
DTIME	Inner macro of DCUST macro	Manager Products	-
DYRCOM1	COMMAND-STREAM member	DYR-TE08	-
DYRCOM2	COMMAND-STREAM member	DYR-TE08	-
DYRCOM3	COMMAND-STREAM member	DYR-TE08	-
DYRCOM4	COMMAND-STREAM member	DYR-TE08	-
DYRCOM5	COMMAND-STREAM member	DYR-TE08	-
EFABBAL	Argument block for BAL functions	CMR-UD05, DYR-TE00	-
EFABCOB	Argument block for COBOL functions	CMR-UD05, DYR-TE00	-
EFABPLI	Argument block for PLI functions	CMR-UD05, DYR-TE00	-
FMTAD	FORMAT member	DSR-UD30	-
FMTAS	FORMAT member	DSR-UD30	-
FMTB	FORMAT member	DSR-UD30, DSR-PH02	-
FMTDD	FORMAT member	DSR-UD30	-
FMTDS	FORMAT member	DSR-UD30	-
FMTED	FORMAT member	DSR-UD30, DSR-EM10	-
FMTENT	FORMAT member	DSR-UD30	-
FMTES	FORMAT member	DSR-UD30, DSR-EM10	-
FMTFD	FORMAT member	DSR-UD30, DSR-PH10	-
FMTFS	FORMAT member	DSR-UD30, DSR-PH10	-
FMTID	FORMAT member	DSR-UD30	-
FMTIS	FORMAT member	DSR-UD30	-
FMTLD	FORMAT member	DSR-UD30	-
FMTLS	FORMAT member	DSR-UD30	-

Table 1. Contents of Dataset MP.CMSRCE

Member Name	Description	Selectable Unit/ Product	Generated Load Module
FMTND	FORMAT member	DSR-UD30, DSR-UD31	-
FMTNS	FORMAT member	DSR-UD30, DSR-UD31	-
FMTPD	FORMAT member	DSR-UD30, DSR-UD31	-
FMTPS	FORMAT member	DSR-UD30, DSR-UD31	-
FMTQ	FORMAT camber	DSR-UD30, DSR-PH01	-
FMTRD	FORMAT member	DSR-UD30	-
FMTRS	FORMAT member	DSR-UD30	-
FMTUD	FORMAT member	DSR-UD30	-
FMTUS	FORMAT member	DSR-UD30	-
FMTUSE	FORMAT member	DSR-UD30	-
FMTVD	FORMAT member	DSR-UD30	-
FMTVS	FORMAT member	DSR-UD30	-
LBUF1	Tailoring macro	DesignManager	DSR00
LOPT1	Tailoring macro	DesignManager	DSR99
MPDX1	DictionaryManager source module	DictionaryManager	MPDX1
MPLFD	Macro used to define User Defined Functions	CMR-UD05, DYR-TE00	-
MPLUF	User Defined Function source module	CMR-UD05, DYR-TE00	MPLUF
MPLX1	Logon Exit source module	CMR-SC05	MPLX1
MPSQL	SQL/DS database access module	DYR-TI12	-
SQLPLT	Format member	DSR-UD30, DSR-PH01	-
SQLPRV	Format member	DSR-UD30, DSR-PH01	-
SQLREP	Format member	DSR-UD30, DSR-PH01	-
SQLVIEW1	SQL import views	DYR-T132	-
SQLVIEW2	SQL import views	DYR-T132	-

Installation Macros

How to Tailor the Manager Products Software using Installation Macros

A number of macros are supplied which allow you to tailor your Manager Products software to your installation's own requirements. These macros are supplied in the MP-CMSRCE dataset on the installation tape. MP.CMSRCE must be copied to disk (as described in Chapter 3, "Copying Dataset MP.CMSRCE," on page 28) for tailoring to take place.

Tailoring your Manager Products using these macros has a global effect on all users and should not therefore be undertaken except on the express instructions of the systems administrator.

For any macros, if the supplied default values of all the keywords listed in the macro's specifications are acceptable, no action need be taken in respect of the macro subsequent to installation. If any values are to be changed, the macro must be submitted to the Assembler, with required values declared for those keywords whose values are to be changed; it is not necessary to declare any keywords whose default values are acceptable. Keywords with their new values can be declared in any order.

For each macro to be used, you must create a source member on the mini-disk containing the Manager Products software. Each source member must have a file type of ASSEMBLE. Each such source member is assembled using the CMS ASSEMBLE command and the resulting text file will overwrite the default version supplied by ASG. This is the required CMS command:

```
GLOBAL MACLIB maclib-name
ASSEMBLE file-name
GLOBAL MACLIB
```

where:

maclib-name is the name of the MACLIB (see Chapter 3, "Step 1 - Copy Datasets to Disk," on page 27).

file-name is the name of the CMS file to be assembled. This must be the same as the name of the module generated when the source member is assembled.

The file can be created by using XEDIT. The required source statements are as follows:

```
macro keyword=value<,keyword=value...>
END
```

where:

macro is the name of the macro whose keyword values are to be changed.

keyword is any keyword from the list in the macro's specifications.

value is any of the alternative values permitted for the keyboard. Where expressly stated in the specification of a keyword, value can alternatively be declared as a sub-list of values, for example:

```
keyword=(value<,value>...)
```

The sub-list must not exceed 255 characters, including the parentheses and commas. If the macro declaration covers more than one line, then the standard rules for coding Assembler statements apply; that is, the continuation character * must be in column 72 and continuation lines must start in column 16.

You can re-tailor Manager Products at any time. Any re-tailoring undertaken changes the ASG-supplied default values, not your tailored values. For this reason, any changes previously made which you wish to retain should be repeated in the re-tailoring process.

Manager Products Installation Macros

Table 2. Installation Macros Available with Manager Products

Macro Name	Purpose	Module Generated	Relevant Selectable Unit
DCHAR	Enables the character set for all Manager Products output to be altered to conform to an installation's particular requirements.	DMI95	Any
DCUST	Enables the following to be tailored: • Number of lines per printed page of printed output (keyword PAGE) • Number of buffers to be allocated to the buffer pool for the dictionary (keywords IBUF, SBUF, DBUF, IBUFC, SBUFC, DBUFC) and the MP-AID (keywords MBUF and MBUFC) • The use of Manager Products updating commands (keywords UPMPAID and UPDICT). • The input and output formats for date and time elements • Certain system parameters for running Manager Products in OS and DOS environments • Certain system parameters for running Manager Products under the IBM CICS teleprocessing monitor	DMU09	Any
DLOG	Enables the logging facility to be tailored.	DML99	Any
MPLFD	Enables User Defined Functions to be defined.	MPLUF	CMR-UD05, DYR-TE00

DCHAR

The macro DCHAR enables the character set for all Manager Products output to be altered to conform to an installation's particular requirements. Output is translated according to a 256-byte translation table. Each byte in the table is the character to be output for a byte whose value is

equivalent to the displacement of this character from the beginning of the table. The default translation table (which contains the PL/I 60-character set) is shown below:

Default Translation Table for Macro DCHAR (A Matrix Giving Hexadecimal Displacement of Characters)

Character	Character Position (b)															
Position (a)	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2																
3																
4												.	<	(	+	¦
5	&										!	£	*	)	;	¬
6	-	/										,	%	_	>	?
7											:	#	@	'	=	"
8		a	b	c	d	e	f	g	h	i						
9		j	k	l	m	n	o	p	q	r						
A			s	t	u	v	w	x	y	z						
B																
C		A	B	C	D	E	F	G	H	I						
D		J	K	L	M	N	O	P	Q	R						
E			S	T	U	V	W	X	Y	Z						
F	0	1	2	3	4	5	6	7	8	9						

where:

a is the first digit in the 2-digit hexadecimal displacement.

b is the second digit in the 2-digit hexadecimal displacement.

Any character not present in this table is translated to a space. If the default table is acceptable, no action need be taken in respect of DCHAR.

The hatched areas in the table represent hexadecimal codes for which the character output will depend on the hardware and software which you are using at your installation.

The macro DCHAR has three keywords, LOWER, MODIFY, and MODIFY2. The purpose of the LOWER keyword is to indicate how lower-case alphabetic characters are to be handled. The alternatives are:

LOWER=NO	Translate lower-case characters to spaces.
LOWER=YES	Translate lower-case characters to lower-case characters (that is, print them unchanged).
LOWER=UPPER	Translate lower-case characters to corresponding upper case characters.

The default value for LOWER is YES.

The MODIFY keyword allows individual characters in the default translation table to be changed. It has the form:

```
MODIFY (abyy<,abyy>...)
```

where *ab* is the 2-digit hexadecimal displacement in the table at which the character, whose 2-digit hexadecimal representation is *yy*, is to be placed. The parentheses can be omitted if there is only one entry in the MODIFY sub-list.

The sub-list for the MODIFY keyword must not exceed 255 characters, including the parentheses and commas. If this is not sufficient to make the required changes to the character set, you may use the MODIFY2 keyword to create an additional sub-list of up to 255 characters.

To change the character set for Manager Products output, the procedure described in "How to Tailor the Manager Products Software using Installation Macros" on page 39 must be followed for DCHAR, specifying the required values for the three keywords as required.

Examples:

To translate lower-case alphabetic characters to the corresponding upper characters and to include square brackets in the set of printable characters, the macro call is:

```
DCHAR LOWER=UPPER,MODIFY=(ADAD,BDBD)
```

To remove the 'logical' not character from the set of printable characters, the macro call is:

```
DCHAR MODIFY=5F40
```

The declared keywords can be in any order.

The printable character set can be re-tailored at any time, by submitting a job as in the foregoing illustrations. If the set is re-tailored, the tailoring is again of the set as supplied, not of the set as last altered.

The System Administrator's SET CHARACTER-TRANSLATION OUTPUT command allows the values in the Default Output Translation Table to be temporarily modified during a Manager Products run. It also allows translation of hexadecimal codes to be different for printed output and screen output.

Note:

The System Administrator can also enable translation of input hexadecimal codes using the SET CHARACTER-TRANSLATION INPUT command.

DCUST

Table 3. The Macro DCUST: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
AUTORW	Whether or not READ/WRITE access to the MP-AID via Auto Logon is allowed.	NO	YES
DBUF	The number of buffers to be allocated to the Data Entries dataset buffer pool in a non-CICS environment (Note 1).	4	2 to 32,000
HANGUL	Specifies a default HANGUL environment.	NO	YES
IBUF	The number of buffers to be allocated to the Index dataset buffer pool in a non-CICS environment (Note 1).	4	2 to 32,000
IOTYPE	Applicable in DOS environments only (Note 10).		
KANJI	Specifies a default KANJI environment. The KANJI environment can subsequently be changed by the systems administrator's SET KANJI-MODE command.	None	IBM, HIT, or FUJ (Note 12)
LOGEXIT	Whether the LOGON EXIT is to be invoked in all environments or restricted to use in full screen interactive environments only.	ALL	ONLINE
MAXLOG	The maximum number of logon attempts permitted in a full screen interactive environment. A value of 0 specifies no limit.	0	1 to 30,000
MBUF	The number of buffers to be allocated to the MP-AID buffer pool in a non-CICS environment (Note 1).	3	2 to 32,000
PAGE	The number of lines per page of printed output.	60	2 to 255

Table 3. The Macro DCUST: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
PCBXREF	Selects cross reference integrity checking during the encoding of any INS Program Communication Block (PCB) member types.	YES	NO—if the resource requirements of the checks is found to be restrictive, in which case you take responsibility for the correct selection end sequence of segments included in the PCB.
PDSQNAM	Applicable in OS environments only.		
SBUF	The number of buffers to be allocated to the Source dataset buffer pool in a non-CICS environment (Note 1).	2	2 to 32,000
SYSxxxx	Applicable in DOS environments only.		(Note 10)
UPDICT	Whether dictionary updating commands are to be accepted (Note 2).	YES	NO
UPMPAID	Whether MP-AID updating commands are to be accepted (Note 3).	YES	NO
Parameters limiting the number of blocks preformatted when creating or reloading a dictionary or an MP-AID using Manager Products commands (Note 11):			
DNUM	For the Data Entries dataset.	0	An integer of not more than 9 digits.
INUM	For the Index dataset.	0	An integer of not more then 9 digits.
LNUM	For the Log dataset.	0	An integer of not more than 9 digits.
MNUM	For the MP-AID.	0	An integer of not more than 9 digits.
RNUM	For the Recovery dataset.	0	An integer of not more than 9 digits.
SNUM	For the Source dataset.	0	An integer of not more than 9 digits.
These keywords are for use in CICS environments only:			
DBUFC			
IBUFC			
ITRAN			
MBUFC			

Table 3. The Macro DCUST: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
RETRY			
RTRAN			
SBUFC			
TEMPST			
TSKEY			
Parameters controlling date and time formats:			
DISEP	Whether a separator character is to be used between date elements on input.	YES	NO
DOSEP	Whether a separator character is to be used between date elements on output and, if so, what that character is to be.	' ' (space)	Empty or required separator character. If the required character is a space or a comma, it should be enclosed by ' ' .
IDATE	The order in which Day (D), Month (M), and Year (Y) are to be input.	DMY	Required order of D, M, and Y. M is optional (Note 5).
IDAYL	Fixed length of day field for input when separator not specified.	None	2 or 3 (Note 7)
IDM1 through IDM12	Name (or names) to be used on input for the month indicated by the numeric portion of the parameter.	(Note 9)	Any undelimited character string of not more than 20 characters identifying the month, or a sub-list of such strings (Note 8).
IMONL	Fixed length of month field for input when separator not specified.	None	2 (Note 7)
ITIME	The order in which Hour (H), Minute (M), Second (S), and AM/PM (X) are to be input.	HMS	Required order of H, N, S, and X. S and X are optional (Note 4).
IYRL	Fixed length of year field for input when separator not specified.	None	2 or 4 (Note 7)
Parameters controlling date and time formats (continued):			

Table 3. The Macro DCUST: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
MAX2000	For input dates, specifies the maximum value, for a year entered as 2 digits *xx*, to be converted to a 4 digit year in the form 20*xx*. Two digit years entered above this value are converted as 19*xx*.	49	10 to 90
ODATE	The order in which Day (D), Month (M) or Literal (L), and Year (Y) are to be output.	DLY	Required order of D, M or L, and Y. M and L are mutually exclusive (Note 5 & Note 6).
ODAYL	Length of the day field for output.	2	3
ODM1	Name to be used on output for the month indicated by the numeric portion of the parameter.	JAN	Any undelimited character string identifying the month, or empty. If empty, the month number is used.
ODM2		FEB	
ODM3		MAR	
ODM4		APR	
ODM5		MAY	
ODM6		JUN	
ODM7		JUL	
ODM8		AUG	
ODM9		SEP	
ODM10		OCT	
ODM11		NOV	
ODM12		DEC	
OMONL	Length of month field for output.	0 (Note 6)	2
OTIME	The order in which Hour (H), Minute (M), Second (S), and AM/PM (X) are to be output.	HMS	Required order of H, M, S, and X. S and X are optional (Note 4).

Table 3. The Macro DCUST: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
OYRL	Length of year field for output.	4	2
TISEP	Whether a separator character is to be used between time elements on input.	YES	NO
TOSEP	Whether a separator character is to be used between time values on output and, if so, what that character is to be.	. (period)	Empty or required separator character. If the required character is a space or a comma it should be enclosed by ' '.

Notes

1. The keywords IBUF, DBUF, and SBUF can be overridden in any particular run by including the clauses IBUF *n*, DBUF *n* and SBUF *n* respectively in the DICTIONARY command, where *n* is the number of buffers to be allocated to the dataset buffer pool. At least 2 buffers must be specified for each of these 3 datasets.

   The number of buffers allocated to the MP-AID buffer pool can be re-specified at any time by issuing the command MP-AID BUFFERS *n* where *n* is the number of buffers required. At least 2 buffers must be specified for the MP-AID.

   A buffer is defined as an area of virtual storage into which physical blocks from a given dataset are read. A separate buffer pool is created for each of the Index, Source, and Data Entries datasets and the MP-AID, and must consist of at least 2 buffers. For the effects of varying the size of a buffer pool, see *ASG-Manager Products Performance Tuning*.

2. If the value of UPDICT is YES, the DICTIONARY command opens a dictionary to allow updating. If NO is declared, the DICTIONARY command opens a dictionary for read-only access. This provides global control over an entire installation. If you have the systems administrator's Environmental Control Facility (CMR-SCO5) installed, then the systems administrator may override the value of UPDICT by using the SET DICTIONARY-UPDATES command during a Manager Products run.

   If logging of non-updating commands and/or messages is to be applied to a dictionary, then updating of the dictionary must be allowed.

3. If the value of UPMPAID is set to NO then all MP-AID updates are inhibited. If you have the systems administrator's Environmental Control Facility (CMR-SCO5) installed, then the systems administrator may override the value of UPMPAID by using the SET MPAID-UPDATES command during a Manager Products run.

4. If X is specified for ITIME or OTIME, hours input or output respectively will be in the form of 1 to 12; otherwise the 24-hour clock will be used.

5. If M is omitted from IDATE, or M and L are both omitted from ODATE, the relevant date will be assumed to be in Julian format; that is, DDD. In this case if separators are not required, the value of IDAYL must be 3.

6. If M is specified as the value assigned to ODATE, then no literals can be output. In this case the value assigned to OMONL must be 2; the values assigned to ODM1 thru ODM12 will be ignored and the month numbers used.

   If L is specified in the value assigned to ODATE, then literals can be output. If literals are to be output, the value assigned to ODM1 thru ODM12 must contain the character strings required.

7. A value must be specified for this parameter when DISEP=NO.

8. For IDM1 to IDM12 any number of character strings may be specified for each month, within an overall limit of 255 characters for the sub-list, including parentheses and separators.

9. The default value of an IDM*n* parameter is the same value as the specified or default assigned to the corresponding ODM*n* parameter.

10. This parameter must not be coded in OS environments.

11. These parameters are not normally used except in special environments such as VSPCC where it is not otherwise possible to limit the number of physical blocks formatted by Manager Products.

12. The KANJI keyword is used to define one of 3 Kanji Modes where a Kanji Mode represents a Kanji supporting hardware/software environment.

    If KANJI is set to IBM, then the Kanji Mode is defined as IBM.

    If KANJI is set to HIT, then the Kanji Mode is defined as HITACHI.

    If KANJI is set to FUJ, then the Kanji Mode is defined as FUJITSU.

    Each Kanji Mode supported by Manager Products recognizes a different set of Shift-Out (SO) and Shift-In (SI) characters.

    Once a Kanji Mode is set, Kanji character strings, enclosed within SO and SI characters, may be input within delimited character strings.

Examples

```
DCUST PAGE=50,IBUF=10,DBUF=20
DCUST IDM1=(JAN,'01',JANUARY)
DCUST IDM1=JANUARY
```

DLOG

Table 4. The Macro DLOG: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
COMTYPE	Whether the updating commands or all commands are to be logged.	UPDATE	ALL (Note 1)
MSGLEV	Severity levels of Manager Products messages that are to be logged.	None (Note 4)	(Note 1 and Note 2)
MSGNO	Identifying numbers of Manager Products messages that are to be logged.	None (Note 4)	(Note 1 and Note 2)
MAXEXCP	The number of input/output accesses needed for a full roll forward, at which a warning message is issued when a dictionary is opened.	50,000	Up to 16,777,215

Notes

1. Alternative values for this parameter have no effect if the ControlManager Audit and Security facility, selectable unit CMR-DD3, is not installed.

2. This value can be declared as one of the letters I, W, E, S, or C, or as a sub-list of any combination of those letters. Any message output by a Manager Products with an alphabetic suffix to its identifying number that corresponds to a letter declared in this value is logged. (For the significance of these alphabetic suffixes, see *ASG-Manager Products Message Guide*.)

3. This value can be the identifying number of a Manager Products message, or a sub-list of such numbers. Message numbers can be declared with or without leading zeros. If a sub-list is declared, it must not exceed 255 characters including parentheses and commas.

4. If no values for the keywords MSGLEV and MSGNO are declared, no messages are logged.

5. Certain commands are not logged, for example, AUDIT commands.

6. Messages arising from non-loggable commands are not logged.

MPLFD

Table 5. The Macro MPLFD: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
TYPE	The type of entry to be made to source module MPLUF.	None (Note 1)	INITIAL, ENTRY, FINAL (Note 1)
NAME	The name of the User Defined Function you wish to define.	None (Note 2)	(Note 2)
EP	The name or entry point of a module if this differs from that of the User Defined Function as specified in the NAME keyword.	None (Note 3)	(Note 3)

Notes

1. TYPE=INITIAL and TYPE=FINAL define the beginning and end of the list of function names. These two entries are supplied in source module MPLUF and do not therefore need to be specified by the user.

   A TYPE=ENTRY statement is required for each User Defined Function you wish to define.

2. A name must be specified for each User Defined Function you wish to use and can be up to 8 characters long.

3. EP (Entry Point) is an optional additional operand. The name specified can be up to 8 characters long.

DataManager Installation Macros

Introduction

Various DataManager installation macros are supplied. Further details can be found in publications referred to below.

Table 6. DataManager Installation Macros

Macro Name	Module Generated	Facility Name	Relevant Selectable Unit
DCONV	DFU07	Automation of Set Up	DMR-AS1, DMR-AS2
DGCOB	DFU11	This publication ("DGCOB" on page 53)	DMR-SL1, DMR-SL5, DMR-SL6, DMR-SL9
DGPLI	DFU12	This publication ("DGPLI" on page 58)	DMR-SL2, DMR-SL5, DMR-SL6, DMR-SL9
DGBAL	DFU10	This publication ("DGBAL" on page 61)	DMR-SL3, DMR-SL5, DMR-SL6, DMR-SL9
DGREC	DFU14	This publication ("DGREC" on page 64)	DMR-SL1, DMR-SL2, DMR-SL3

Table 6. DataManager Installation Macros

Macro Name	Module Generated	Facility Name	Relevant Selectable Unit
DGADA	DFU16	ADABAS Interface	DMR-SL6
DGDBD	DIL88	IMS (DL/I) Interface	DMR-SL5
DGPSB	DIL89	IMS (DL/I) Interface	DMR-SL5
DGSCOB	DIL99	IMS (DL/I) Interface	DMR-SL5
DGSPLI	DIL98	IMS (DL/I) Interface	DMR-SL5
DGSBAL	DIL97	IMS (DL/I) Interface	DMR-SL5
DGSREC	DIL96	IMS (DL/I) Interface	DMR-SL5
DDS2K	DYD99	SYSTEM 2000/80 Interface	DMR-SL9
DGS2K	DYD11	SYSTEM 2000/80 Interface	DMR-SL9
DGTOT	DFU13	TOTAL Interface	DMR-SL4
DGMIV	DFU15	MARK IV Interface	DMR-SL7

The DCONV installation macro is used with the Automation of Set-Up (COBOL) facility (DMR-AS1) or the Automation of Set-up (PL/I) facility (DMR-AS2). For details see the appropriate section in *ASG-DataManager Automation of Set-up*.

Source Language Generation Macros

The installation macros DGCOB, DGPLI, DGBAL, and DGREC allow the Controller to tailor output formats and options to suit a particular installation's requirements.

DGCOB can be used to tailor the generation of COBOL source language data descriptions and of record layouts generated in association with COBOL source language descriptions.

DGPLI can be used to tailor the generation of PL/I source language data descriptions and of record layouts generated in association with PL/I source language data descriptions.

DGBAL can be used to tailor the generation of Basic Assembler Language source language data descriptions and of record layouts generated in association with Basic Assembler Language source language data descriptions.

DGREC can be used to tailor the generation of record layouts, when no associated source language data descriptions are being generated.

DGCOB

The keywords of the macro DGCOB are defined in the following table. The macro assembles as module DFU11.

Table 7. The Macro DGCOB: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
ACHAR	The hexadecimal values of any additional characters that are to be accepted for output in names produced by the Source Language Generation facility, to enable characters not in the standard source language character set to be output (Note 1).	None	Any valid hexadecimal value, or a sub-list of such values.
ACHAR2	As for ACHAR (above) to allow additional valid values (Note 7).	None	As for ACHAR (above).
ACSMETH	The type of file generated by PRODUCE.	BPAM	DSAM
ALIAS	Whether COBOL specific aliases are to be generated instead of member names.	NO	YES (Note 2)
ATRUNK	The character part of an ALPHABETIC ITEM FILLER name.	ALPHA-FILLER	Name (Note 3)
AUTOCHK	Check for and convert fillers.	YES	NO
BINSIGN	Whether items defined as BINARY (COBOL COMP) are to be signed implicitly in the PICTURE clause.	YES	NO
BTRUNK	The character part of a BINARY ITEM FILLER name.	BIN- FILLER	Name (Note 3)
COBOL2	Whether COBOL II is to be generated instead of VS COBOL.	NO	YES
COL01	Starting column for 01 level number.	8	Up to 99
COL2ND	Starting column for second level number.	12	Up to 99
COLCOND	Whether level 88 statements generated from CONDITION-NAME clauses are to be output in a fixed position or in a position relative to the generated conditional variable.	OFFSET	FIXED
COLCPOS	The offset of CONDITIONAL (level 88) clauses (if COLCOND set to OFFSET).	1	0 to 10
COLCPOS	The starting column for CONDITIONAL (level 88) clauses (if COLCOND set to FIXED).	15	12 to 40
COLMAIN	Starting column of PICTURE clause and VALUE clause.	41	Up to 99

Table 7. The Macro DGCOB: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
COLNOTE	Starting column for NOTEs and DESCRIPTIONs that are printed as comments (column 7 always contains an asterik).	12	8 to 11 13 to 72
COLSEQ	Starting column of line sequence number.	1	Up to 99
COLSUBS	Starting column of statement elements after PICTURE clause (for example, SYNC).	56	Up to 99
COMP	Whether BINARY items are to be generated as COMPUTATIONAL instead of BINARY and/or PACKED-DECIMAL items are to be generated as COMPUTATIONAL-3 instead of PACKED-DECIMAL (applies only if COBOL2=YES).	YES	NO
CONCARD	Whether a control card is to be produced.	YES	NO
COND88	Whether level 88 statements are to be generated from CONDITION-NAME clauses (with associated RANGE or IS clauses) found in ITEM data definition.	YES	NO
DDNAME	Default library name.	GENLIB	Name
DECOMMA	Whether decimal comma is to be generated instead of decimal point.	NO	YES
DESC	Maximum number of character strings of DESCRIPTION clauses used to generate comments.	0	Up to 32,767 or ALL
DNOTES88	Whether level 88 statements are to be found in NOTE clauses.	NO	YES
DTRUNK	The character part of a PACKED-DECIMAL ITEM FILLER name.	DEC-FILLER	Name (Note 3)
FILESUF	The suffix appended to file-name where 01 data-name is automatically generated.	-REC	Suffix (Note 4)
FTRUNK	The character part of a FLOATING-POINT ITEM FILLER name.	FLOAT-FILLER	Name (Note 3)
GEN	Whether both FD and 01 levels are to be generated from FILE members, or FD only, or 01 only.	ALL	FD or 01
GFNL	Length of number part of GROUP FILLER name.	5	4 to 15
GTRUNK	Value of name part of GROUP FILLER name.	GROUP-FILLER	Name (Note 3)
IFNL	Length of number part of ITEM FILLER name.	5	4 to 14

Table 7. The Macro DGCOB: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
INCLEV	Level number increment.	2	Up to 99
INCLEV0	Whether level numbering increments are to begin from zero or from 01. Default value gives level numbers of 01,02,04... Alternative value gives level numbers of 01,03,05...	YES	NO
INCRSEQ	Line sequence number increment.	10	Up to 32,767
INITVAL	Whether VALUE clauses are to be generated from ITEM date definitions.	NO	YES
KEYABB	Whether permitted keyword abbreviation is to take place.	YES	NO
KNOWNAS	Whether local-names from KNOWN-AS clauses are to be generated instead of member names.	NO	YES (Note 2)
LENSEQ	Length of sequence number.	6	Up to 9
LIBCC	The format of the control card output as the first card of a QSAM file (unless overridden by an ONTO CLAUSE).	(see *ASG-Manager Products Source Language Generation*)	A delimited character string of 1 to 72 characters (including question mark for which a generated library name is substituted).
MAXFILE	Maximum length of any file data structure.	1,048,575	Any value up to the limit for MAXLEN below.
MAXLEN	Maximum length of any data structure.	16,777,215	Up to 2,147,483,647
MAXSYM	Maximum number of PICTURE symbols before replacement with a repetition factor takes place.	3	From 1 to 30
MEMLEN	Maximum length of library-name.	8	Up to 16
NEZEROS	Whether to assume a default value for numeric edited items of ZERO (Yes) or SPACES (No). This option is only relevant where the item has a picture clause including editing symbols and no contents clause is present and initial values are required.	YES	NO

Table 7. The Macro DGCOB: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
NOTE	Maximum number of character strings of NOTES clauses to be used to generate comments.	0	Up to 32,767 or ALL
NTRUNK	The character part of a NUMERIC-CHARACTER ITEM FILLER name.	NC- FILLER	Name (Note 3)
NUMSIGN	Whether items defined as NUMERIC (COBOL DISPLAY) are to be signed implicitly in the PICTURE clause.	NO	YES
OFFSUBS	Number of spaces between subsequent level numbers.	2	Up to 99
PCKSIGN	Whether items defined as PACKED-DECIMAL (COBOL COMP-3) are to be signed implicitly in the PICTURE clause.	YES	NO
QUOTES	Whether generated non-numeric literals are to be enclosed within double (") or single (') quotation marks.	DOUBLE	SINGLE
RECBOX	Whether boxing of record layouts printout required.	YES	NO
RECCON	Not yet implemented.		
RECFMGEN	Whether the RECORD FORMAT clause is generated only for UNDEFINED blocking.	NO	YES (Note 6)
RECPGSI	Number of lines per printed page in record layouts.	SYSTEM (Note 5)	1 to 255
RECPIC	Whether repository PICTUREs required in record layouts.	YES	NO
RECPOS	Whether start positions are to be generated (in record layouts) instead of offsets.	NO	YES
RECSP	Number of space lines between each item in record layout (applies only if RECBOX=NO).	1	Up to 255
REDFILL	Whether unequal redefining members are to be padded with fillers.	YES	NO
RNDBIN	Whether 1- and 3-byte binary items are to be rounded up.	YES	NO
RNDBIT	Whether bit string fields are generated with byte alignment (see *ASG-Manager Products Source Language Generation*).	YES	NO
SEQNO	Whether line sequence numbering required.	NO	YES

Table 7. The Macro DGCOB: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
SPACING	Number of spaces between statement elements.	1	Up to 64
ULABNAM	Name generated for use in FD LABEL RECORDS clause when USER-LABELS specified.	LABEL-	Name
VALUE	Whether a VALUE of SPACE(S) or ZERO(S) is to be generated if an ITEM does not have a CONTENTS clause, instead of omitting the VALUE clause.	YES	NO

Notes

1. The standard Source Language Generation facility output character set for COBOL data descriptions conforms to that defined for American National Standard COBOL for the data division. The character set can be extended to allow non-standard characters to be put in names, by entering the hexadecimal value of each required character as a value to ACHAR. The user should ensure that any extra characters that are added to the output character set in this way are used only in ways that are permitted by the software with which Manager Products are used.

2. If ALIAS=YES and KNOWNAS=YES both apply, then when a data name is generated for a member that has an ALIAS clause and is subject to a containing member's KNOWN-AS clause, the KNOWN-AS local name takes precedence.

3. Name defines part of a member name. It must be stated within single quotes, must not be more than 16 characters in length and must conform to Manager Products rules for member names stated in InfoBank panel MPLANG5100. The values declared for the keywords ATRUNK, BTRUNK, FTRUNK, GTRUNK, and NTRUNK should correspond to those of the same keywords defined in association with the import from COBOL function, if that function is also used.

4. The value of the keyword FILESUF must be a suffix acceptable in COBOL of up to 29 characters enclosed in single quotes.

5. The option RECPGSI=SYSTEM means that the number of lines printed per page is the value of the PAGE keyword in the DCUST macro.

6. If RECFMGEN=NO, RECORD FORMAT is generated only if the member is defined with a BLOCKING clause of UNDEFINED. If RECFMGEN=YES, RECORD FORMAT is generated for all values of BLOCKING.

7. The format of ACHAR operands limits input to 127 values. The ACHAR2 keyword allows for a second set of up to 127 values to be specified, if required.

DGPLI

The keywords of the macro DGPLI are defined in the following table. The macro assembles as module DFU12.

Table 8. The Macro DGPLI: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
ACHAR	The hexadecimal values of any additional characters that are to be accepted for output in names produced by the Source Language Generation facility, to enable characters not in the standard source language character set to be output (Note 1).	None	Any valid hexadecimal value, or a sub-list of such values.
ACHAR2	As for ACHAR (above) to allow additional valid values (Note 6).	None	As for ACHAR (above).
ACSMETH	The type of file generated by PRODUCE.	BPAM	QSAM
ALIAS	Whether PL/I specific aliases are to be generated instead of member names.	NO	YES (Note 2)
ATRUNK	The character part of an ALPHABETIC ITEM FILLER name.	ALPHA-FILLER	Name (Note 3)
AUTOCHK	Check for and convert fillers.	YES	NO
BSDSUF	Suffix for generated BASED structured names.	_BASED	Suffix (Note 4)
BTRUNK	The character part of a BINARY ITEM FILLER name.	BIN-FILLER	Name (Note 3)
CHARSET	Whether the 60- or 48-character set is to be used.	60	48
COL01	Starting column for 01 level number.	2	Up to 99
COLMAIN	Starting column for attributes.	41	Up to 99
COLNOTE	Starting column for comments.	41	Up to 99
COLSEQ	Starting column for sequence number.	73	Up to 99
CONCARD	Whether a control card is to be produced.	YES	NO
CTRUNK	The character part of a CHARACTER ITEM FILLER name.	FILLER	Name (Note 3)
DDNAME	Default library name.	GENLIB	Name
DESC	Maximum number of character strings of DESCRIPTION clauses used to generate comments.	0	Up to 32,767 or ALL
DTRUNK	The character part of a PACKED-DECIMAL ITEM FILLER name.	DEC-FILLER	Name (Note 3)

Table 8. The Macro DGPLI: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
FLOATYP	Whether BINARY or DECIMAL FLOAT is to be generated.	BINARY	DECIMAL
FTRUNK	The character part of a FLOATING-POINT ITEM FILLER name.	FLOAT-FILLER	Name (Note 3)
GFNL	The number of digits in the number part of a GROUP FILLER name.	5	Up to 14
GTRUNK	Value of name part of GROUP FILLER name.	GROUP-FILLER	Name (Note 3)
IFNL	The number of digits in the number part of an ITEM FILLER name.	5	Up to 14
INCLEV	Level number increment.	2	Up to 99
INCRSEQ	Sequence number increment.	10	Up to 32,767
INITVAL	Whether INITIAL attributes are to be generated from ITEM data definitions.	NO	YES
KEYABB	Whether keyword abbreviation is required.	YES	YES
KNOWNAS	Whether local-names from KNOWN-AS clauses are to be generated instead of member names.	NO	YES (Note 2)
LENSEQ	Length of sequence number.	8	Up to 9
LIBCC	The format of the control card output as the first card of a QSAM file (unless overridden by an ONTO CLAUSE).	(see *ASG-Manager Products Source Language Generation*)	A delimited character string of 1 to 72 characters (including question mark for which a generated library name is substituted).
MAXLEN	Maximum number of data structures.	2,147,483,647	Up to 2,147,483,647
MEMLEN	Maximum length of library-name.	8	Up to 16
NOTE	Maximum number of character strings of NOTES clauses to be used to generate comments.	0	Up to 32,767 or ALL
NTRUNK	The character part of a NUMERIC-CHARACTER ITEM FILLER name.	NC-FILLER	Name (Note 3)
OFFSUBS	Offset for subsequent level numbers.	2	Up to 99
PNTRSUF	Suffix for generated POINTER names.	_PTR	Suffix (Note 3)

Table 8. The Macro DGPLI: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
RECBOX	Whether boxing of record layouts printout required.	YES	NO
RECCON	Not yet implemented.		
RECPGSI	Number of lines per printed page in record layouts.	SYSTEM (Note 5)	1 to 255
RECPIC	Whether repository PICTUREs required in record layouts.	YES	NO
RECPOS	Whether start positions are to be generated (in record layouts) instead of offsets.	NO	YES
RECSP	Number of space lines between each item in record layout (applies only if RECBOX=NO).	1	Up to 255
REFSUF	PL/I REFER option suffix.	_REFER	Suffix (Note 4)
RNDBIN	Whether 1- and 3-byte binary items are to be rounded up.	NO	YES
RNDBIT	Whether bit string fields are generated with byte alignment (See *ASG-Manager Products Source Language Generation*).	NO	YES
SEQNO	Whether line sequence numbering required.	NO	YES
SPACING	The number of spaces between keywords.	1	Up to 99
VALUE	Whether a value of SPACES or ZEROES, according to data type, is to be assigned when initialization is required and no explicit initialization requirement is found.	YES	NO

Notes

1. The standard Source Language Generation facility output character set for PL/I is the PL/I 60-character set. This character set can be extended to allow nonstandard characters to be output in names, by entering the hexadecimal value of each required character as a value to ACHAR. The user should ensure that any extra characters that are added to the output character set in this way are used only in ways that are permitted by the software with which Manager Products is used.

2. If ALIAS=YES and KNOWNAS=YES both apply, then when a data name is generated for a member that has an ALIAS clause and is subject to a containing member's KNOWN-AS clause, the KNOWN-AS local-name takes precedence.

3. Name defines part of a member name. It must be stated within single quotes, must not be more than 16 characters in length, and must conform to Manager Products rules for member names stated in InfoBank panel MPLANG5100. The values declared for the keywords ATRUNK, BTRUNK, CTRUNK, DTRUNK, FTRUNK, GTRUNK, and NTRUNK should correspond to those of the same keywords defined in association with the Automation of Set Up facility for the generation of members from PL/I source statements, if that facility is also used.

4. Suffix must be a suffix acceptable in PL/I. It must be given within single quotes.

5. The option RECPGSI=SYSTEM means that the number of lines printed per page is the value of the PAGE keyword in the DCUST macro.

6. The format of ACHAR operands limits input to 127 values. The ACHAR2 keyword allows for a second set of up to 127 values to be specified, if required.

DGBAL

The keywords of the macro DGBAL are defined in the following table. The macro assembles as module DFU10.

Table 9. The Macro DGBAL: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
ACHAR	The hexadecimal values of any additional characters that are to be accepted for output in name produced by the Source Language Generation facility, to enable characters not in the standard source language character set to be output (Note 1).	None	Any valid hexadecimal value, or a sub-list of such values.
ACHAR2	As for ACHAR (above) to allow additional valid values (Note 5).	None	As for ACHAR (above).
ACSMETH	The type of file generated by PRODUCE.	BPAM	QSAM
ALIAS	Whether Assembler specific aliases are to be generated instead of member names.	NO	YES (Note 2)
ATRUNK	The character part of an ALPHABETIC ITEM FILLER name.	ALPHA- FILLER	Name (Note 3)
AUTOCHK	Whether to check for and convert fillers.	YES	NO
BTRUNK	The character part of a BINARY ITEM FILLER name.	BIN- FILLER	Name (Note 3)
COLMAIN	Starting column for operation code.	10	Up to 99
COLNOTE	Starting column for comments.	41	Up to 99
COLSEQ	Starting column for sequence number.	73	Up to 99
COLSUBS	Starting column for operand.	16	Up to 99
CONCARD	Whether a control card is to be generated.	YES	NO

Table 9. The Macro DGBAL: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
CTRUNK	The character part of a CHARACTER ITEM FILLER name.	FILLER	Name (Note 3)
DDNAME	Default library name.	GENLIB	Name
DTRUNK	The character part of a PACKED-DECIMAL ITEM FILLER name.	DEC-FILLER	Name (Note 3)
EPATSUF	Edit pattern suffix.	EP	*xx*
EPPROD	Whether edit-patterns are to be generated from the PRODUCE command.	NO	YES
EQUATE	Whether EQU statements are to be generated from CONDITION-NAME clauses (with associated IS clauses) found in ITEM data definitions.	YES	NO
FRDREQD	(Reserved for future use.)	NO	
FRECSUF	(Reserved for future use.)	FD	
FTRUNK	The character part of a FLOATING-POINT ITEM FILLER name.	FLOAT-FILLER	Name (Note 3)
GFNL	The number of digits in the number part of a GROUP FILLER name.	5	Up to 14
GTRUNK	The character part of a GROUP FILLER name.	GROUP-FILLER	Name (Note 3)
IFNL	The number of digits in the number part of an ITEM FILLER name.	5	Up to 14
INCRSEQ	Sequence number increment.	10	Up to 32,767
INITVAL	Whether DC statements are to be generated from ITEM data definitions.	NO	YES
KNOWNAS	Whether local-names from KNOWN-AS clauses are to be generated instead of member names.	NO	YES (Note 2)
LENSEQ	Length of sequence number field.	8	Up to 9
LIBCC	The format of the control card output as the first card of a QSAM file (unless overridden by an ONTO CLAUSE).	(see *ASG-Manager Products Source Language Generation*)	A delimited character string of 1 to 72 characters (including question mark for which a generated library name is substituted).
MAXLEN	Maximum length of any data structure.	16,777,215	Up to 2,147,483,647

Table 9. The Macro DGBAL: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
MEMLEN	Maximum length of library-name.	8	Up to 16
NAMEMAX	Maximum length for data names.	8	1 to 63
NOTE	Maximum number of character strings of NOTES clauses to be used to generate comments.	0	Up to 32,767 or ALL
NTRUNK	The character part of a NUMERIC-CHARACTER ITEM FILLER name.	NC-FILLER	name (Note 3)
RECBOX	Whether boxing of record layouts printout required.	YES	NO
RECCON	Not yet implemented.		
RECPGSI	Number of lines per printed page in record layouts.	SYSTEM (Note 4)	1 to 255
RECPIC	Whether repository PICIUREs are required in record layouts.	YES	NO
RECPOS	Whether start positions are to be generated (in record layouts) instead of offsets.	NO	YES
RECSP	Number of space lines between each item in record layout (applies only if RECBOX=NO).	1	Up to 255
RNDBIN	Whether binary items are to be rounded up.	NO	YES
RNDBIT	Whether bit string fields are generated with byte alignment (see *ASG-Manager Products Source Language Generation*).	YES	NO
SEQNO	Whether sequence numbering required.	NO	YES

Notes

1. The standard Source Language Generation facility output character set for Basic Assembler Language is that defined in the IBM Basic Assembler Language specification. This character set can be extended to allow non-standard characters to be output in names by entering the hexadecimal value of each required character as a value to ACHAR. The user should ensure that any extra characters that are added to the output character set in this way are used only in ways that are permitted by the software with which Manager Products is used.

2. If ALIAS=YES and KNOWNAS=YES both apply, then when a data name is generated for a member that has an ALIAS clause and is subject to a containing member's KNOWN-AS clause, the KNOWN-AS local-name takes precedence.

3. Name defines part of a member name. It must be stated within single quotes, must not be more than 16 characters in length, and must conform to the rules for member names stated in InfoBank panel MPLANG5100.The values declared for the keywords ATRUNK, BTRUNK, CTRUNK, DTRUNK, FTRUNK, GTRUNK, and NTRUNK should correspond to those of the same keywords defined in association with the Automation of Set Up facility for the generation of members from COBOL source statements, if that facility is also used.

4. The option RECPGSI=SYSTEM means that the number of lines printed per page is the value of the PAGE keyword in the DCUST macro.

5. The format of ACHAR operands limits input to 127 values. The ACHAR2 keyword allows for a second set of up to 127 values to be specified, if required.

DGREC

The keywords of the macro DGREC are defined in the following table. The macro assembles as module DFU14.

Table 10. The Macro DGREC: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
ALIAS	Whether RECORD-LAYOUTS specific aliases are to be generated instead of member names.	NO	YES (Note 2)
BASEDMT	For non-ITEMs, whether the repository member type is to be output as the TYPE literal instead of 'GROUP'.	NO	YES
DESC	Maximum number of character strings of DESCRIPTION clauses used to generate comments.	0	Up to 32,767 or ALL
KNOWNAS	Whether local-names from KNOWN-AS clauses are to be generated instead of member names.	NO	YES (Note 2)
MAXLEN	Maximum number of data structures.	2,147,483,647	Up to 2,147,483,647
NOTE	Maximum number of character strings of NOTE clauses used to generate comments.	0	Up to 32,767 or ALL
RECBOX	Whether boxing of record layouts printout is required.	YES	NO
RECCON	Not yet implemented.		
REPGSI	Number of lines per printed page in record layouts.	SYSTEM (Note 1)	1 to 255
RECPIC	Whether repository PICTUREs are required in record layouts.	YES	NO
RECPOS	Whether start positions are to be generated instead of offsets.	NO	YES

Table 10. The Macro DGREC: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
RECSP	Number of space lines between each item in record layouts (applies only if RECBOX=NO).	1	Up to 255
RNDBIN	Whether binary items are to be rounded up.	NO	YES
RNDBIT	Whether bit string fields are generated with byte alignment (see *ASG-Manager Products Source Language Generation*).	YES	NO
WIDEFMT	Whether a wide or a smaller record format is used.	AUTO	YES or NO (Note 3)

Notes

1. The option RECPGSI=SYSTEM means that the number of lines printed per page is the value of the PAGE keyword in the DCUST macro.

2. If ALIAS=YES and KNOWNAS=YES both apply, then when a data name is generated for a member that has an ALIAS clause and is subject to a containing member's KNOWN-AS clause, the KNOWN-AS local-name takes precedence.

3. If WIDEFMT=YES, the wide record format is always used. If WIDEFMT=NO, the smaller format is always used. With the default setting of AUTO, the wide format is used if the record length exceeds 99,999; otherwise the smaller format is used.

DesignManager Installation Macros

Table 11. Installation Macros Available with DesignManager

Macro Name	Purpose	Module Generated	Relevant Selectable Unit
LOPT1	Allows the tailoring of default values assigned to those DesignManager keywords that do not alter the overall size of the DesignManager load module.	DSR99	DSR-DS01
LBUF1	Allows the sizes of various buffers and work areas to be altered. Larger work areas and buffers may be needed for larger logical designs, while smaller areas can be used to reduce virtual storage requirements.	DSR00	DSR-DS01

Note

1. When the LBUF1 macro is assembled a module (text file) called LUBF1 must be created and the DesignManager software must then be relinked. The commands required to achieve this are:

   ```
   TXTLIB DEL DSRTXT CPOO1
   TXTLIB ADD DSRTXT LBUF1
   MPX5 <OVERLAY>
   ```

   OVERLAY should be specified in the MPX5 statement if DesignManager is to be relinked as an overlay module.

LBUF1

Table 12. The Macro LBUF1: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
MTTMAX	The maximum number of entries in the internal member type table (Note 1).	80	Any unsigned integer (Note 1).
WBSIZE	The size of the workbench design area in full words.	64,000	Any unsigned integer (Note 2).
Parameters applicable only if the User Formatted Output facility, DSR-UD30, is installed:			
FMSIZE	The maximum size of the format area in full words (Note 3).	2,500	Any unsigned integer.

Notes

1. The internal member type table identifies those member types which may be referenced by DesignManager. These member types are as follows:

   — VIEWSET

   — USERVIEW

   — ENTITY

   — FORMAT

   — ITEM

   — GROUP

   — COMMAND-STREAM

   — Any relevant User Defined Syntax member types.

   The default value of 80, assigned to the MTTMAX keyword, only needs to be changed if you have:

   - The ControlManager User Defined Syntax facility, selectable unit CMR-UD1,
   - More than 73 relevant User Defined Syntax member types.

2. This work area is used by DesignManager to perform logical database designs. The default size of 64,000 full words is large enough to handle most database designs. The percentage utilization of the workbench design area can be ascertained at any time during a DesignManager run by the input of a SNAPSHOT command. It gives a good indication of whether a larger workbench design area is necessary. It is advisable not to work extensively at a high percentage utilization (above about 90 percent) as this may increase execution time.

3. Users with the User Formatted Output optional additional facility (selectable unit DSR-UD30) who have very large formats in their FORMAT members may have to increase the value of the parameter FMSIZE. This value must be large enough to accommodate the largest format member that will be processed.

   When a format line is loaded into DesignManager, all spaces except those contained within character strings are removed.

LOPT1

Table 13. The Macro LOPT1: Keywords Specifiable on Installation

Keyword	Specifies	Default Value	Alternative Values
ASOPT1	The rules DesignManager uses to decide whether to generate an assumed record (Note 1).	YES	NO
RHSPNM	The prefix name to be used for an assumed data element, generated on the workbench for inclusion as the non-prime data element in an assumed record (Note 2).	ASSUMED	A string of up to 28 characters.
SEPSTR	The characters to be recognized as separators when name reduction takes place in the PREVIEW, DB2 PREVIEW, POPULATE, and DB2 POPULATE commands (Note 3).	- (hyphen) and _ (underscore)	A string of up to 8 characters.
Parameters applicable only if the Enterprise Modeling facility, selectable unit DSR-EM10 is installed:			
LHSPRE	The character used to cause generation of a single default identifier name for an attribute of an entity.	!	Any single character.
RHSPRE	The character used to cause generation of a default identifier name for an attribute of an entity.	@	Any single character.
Parameters applicable only if the SQL/DS Database Design - First Cut Model facility, selectable unit DSR-PH01, or the DB2 Database Design - First Cut Model facility, selectable unit DSR-PH02 is installed:			
SQLCON	The DB2 or SQL/DS continuation character which is required in the output resulting from the PRODUCE SQL or PRODUCE DB2 command.	-(hyphen)	Any continuation character permitted by DB2 or SQL/DS.
XTRACT	The common clause which contains the SQL/DS or DB2 column type within data-element definitions (Note 4).	COMMENT LINE 1	A string of up to 80 characters which should consist of a clause name, optionally followed by LINE and a line number.

Notes

1. If ASOPT1 is set to YES, assumed records are generated according to the rules described in InfoBank panel DSRNET1300, and MVDs (multivalued dependencies) are represented in a network schema ONLY by multivalued associations and/or secondary key associations, as described in InfoBank panels DSRNET1700 and DSRNET1800.

   If ASOPT1 is set to NO, the same rules apply with the following exception.

An assumed record is *not* generated for the right-hand side of the MVD, and the MVD is represented by a hierarchical-m association instead of a multivalued association if (and only if) an MVD satisfies both the following conditions:

- The set of data elements comprising the reduced left-hand side of the MVD is the key of an FD-record or assumed record already present in the Workbench Design Area (WBDA).
- The set of data elements formed by combining the reduced left-hand side and the reduced right-hand side of the MVD is the key of an FD-record or assumed record also present in the WBDA.

If no value is entered for ASOPT1, a default value of YES is assumed.

2. The remainder of the name consists of the assumed record's workbench number.

3. A space character is always recognized and does not have to be specified. The terminator characters. and ; (period and semi-colon), cannot be used as separators.

4. The effect of the XTRACT keyword is nullified if you have the User Formatted Output facility, selectable unit DSR-UD30, and you use the EXTRACT 1 DATA ELEMENTS clause in a FORMAT member to specify the common clause containing the column type.

Tailoring Source Modules

This section provides information on tailoring the source modules supplied on dataset MP.CMSRCE. Details of each source module are given in the following table. If you wish to tailor a particular source module then you should follow these steps:

1. Copy dataset MP.CMSRCE to disk as described in this publication in "Copying Dataset MP.CMSRCE" on page 28.

2. Any source members to be updated should be moved from the CMS MACLIB (created in Chapter 3, "Step 1 - Copy Datasets to Disk," on page 27) to a CMS file created with a module name as specified in the following table and a file type of ASSEMBLE.

3. The appropriate CMS file must then be updated to the user's requirements and subsequently assembled using the CMS ASSEMBLE command. The resulting text file will overwrite the default version as supplied by ASG.

Table 14. Source Modules

Macro Name	Purpose	Module Generated	Relevant Selectable Unit
DMEX1	Input User Exit. If the User Interface facility (CMR-UI1) is installed, ControlManager calls the DMEX1 module before processing each input line. If the user wishes to validate input lines before passing them back to ControlManager (or rejecting them), the supplied version of DMEX1 must be updated, assembled, and link-edited. This replaces the supplied version by the user-defined version.	DMEX1	CMR-UI1
MPDX1	DictionaryManager User Exit. The module MPDX1 is called during translation rule processing when running DictionaryManager. As supplied, it contains coding specific to the translation of a DataManager ITEM PICTURE into the corresponding IDD picture. MPDX1 is supplied as source to allow you to add your own functions or to modify the supplied code to your own needs, in which case the supplied version of MPDX1 must be updated, assembled, and link-edited. This replaces the supplied version by the user-defined version.	MPDX1	DYR-DY01
MPLUF	Procedures Language Source module. The module allows you to specify User-Defined Functions. For further details, refer to Chapter 9, "Step 7 - Linking External Software To Manager Products Software," on page 95.	MPLUF	CMR-UD05, DYR-TE00
MPLX1	LOGON EXIT source module (MPLX1). The Logon Exit facility allows the systems administrator to modify the standard ControlManager Logon procedure. As supplied, MPLX1 will invoke the standard ControlManager Logon procedure.	MPLX1	CMR-SC05

DesignManager Format Members

The User Formatted Output facility, selectable unit DSR-UD30, is documented in *ASG-DesignManager User Formatted Output*. If the User Formatted Output facility is to be installed, no additional installation job control statements are required. However, ASG recommends that you use the special set of FORMAT members until you are sufficiently familiar with the facility to set up your own FORMAT members.

The source definitions for these members are held as members of the partitioned dataset MP.CMSRCE. In order to use them it is necessary to add them to the dictionary as required.

Each member is prefixed by the command:

```
REPLACE member-name;
```

where *member-name* is as given in Table 15 on page 72. The FORMAT members supplied are shown in Table 16.

Table 15. FORMAT Members Supplied for the User Formatted Output Facility

Report Type	FORMAT Member Name for Detail Report	FORMAT Member Name for Summary Report
Data Element Usage Analysis Report	FMTDD	FMTDS
Data-view Report	FMTVD	FMTVS
Design Audit	FMTAD	FMTAS
Intersecting Data Element Report	FMTID	FMTIS
Logical Schema Report	FMTLD	FMTLS
Network Schema Report	FMTRD	FMTRS
Preview/Populate Entity Definitions	FMTENT (see Note)	
Preview/Populate Userview Definitions	FMTUSE (see Note)	
Userview Report	FMTUD	FMTUS
Members only applicable if the Enterprise Modeling facility, selectable unit DSR-EN10, is installed:		
Entity Report	FMTED	FMTES
Members only applicable if the User Printer Graphics facility, selectable unit DSR-UD31, is installed:		
Logical Schema Cluster Plot	FMTPD	FMTPS
Network Cluster Plot	FMTND	FMTNS
Members only applicable if the Load Factor Calculation facility, selectable unit DSR-PH10, is installed:		
Load Factor Analysis Report	FMTFD	FMTFS
Members only applicable if the DB2 Database Design - First Cut Model facility, selectable unit DSRPH02, is installed:		
DB2 Cluster Plot	DB2PLT (see Note)	
DB2 Table Report	DB2REP (see Note)	
Preview/Populate DB2 Definitions	DB2PRV (see Note)	
Produce DB2 Report	FMTB (see Note)	

Table 15. FORMAT Members Supplied for the User Formatted Output Facility

Report Type	FORMAT Member Name for Detail Report	FORMAT Member Name for Summary Report
Members only applicable if the SQL/DS Database Design - First Cut Model facility, selectable unit DSR-PH01, is installed:		
Preview/Populate SQL Definitions	SQLPRV (see Note)	
Produce SQL Report	FMTQ (see Note)	
SQL Cluster Plot	SQLPLT (see Note)	
SQL Table Report	SQLREP (see Note)	

Note:

There is only one type of report available, which is neither a Detail nor a Summary report.

DictionaryManager Command-stream Members

If you have the DictionaryManager Corporate Dictionary Definition Export for IDD facility (DYR-TE08) installed, you may use the DictionaryManager COMMAND-STREAM members supplied on dataset MP.CMSRCE. In order to use them you need to add them to the dictionary as required.

Each member is prefixed by the command:

```
REPLACE member-name;
```

where *member-name* is as given in the Table 16 on page 73. The COMMAND-STREAM members supplied are shown in Table 16.

Table 16. DictionaryManager COMMAND-STREAM Members

MP.CMSRCE Member Name	Command-stream name	Function
DYRCOM1	GEN-CONTAINED-RECORDS	To produce input for the IDD DDDL compiler from IDMS-RECORD members.
DYRCOM2	PRODUCE-IDMS-SCHEMA	To produce input for the IDMS Schema compiler from IDMS-DATABASE members with COBOL-like layouts for IDMS-RECORDS.

Table 16. DictionaryManager COMMAND-STREAM Members

MP.CMSRCE Member Name	Command-stream name	Function
DYRCOM3	GEN-IDMS-SCH	To produce input for the IDMS Schema compiler from IDMS-DATABASE members.
DYRCOM4	GEN-IDMS-SUB	To produce input for the IDMS Subschema compiler from IDMS-SUBSCHEMA members.
DYRCOM5	GEN-IDMS-DMCL	To produce input for the IDMS DMCLcompiler from IDMS-SUBSCHEMA members.

6 Step 4 - Create and Load the MP-AID

Introduction

Before Manager Products can be used, an MP-AID must be created. The MP-AID is the dataset that contains the information used to control the environment in which your Manager Products will operate. A description of the MP-AID and its uses can be found in the *ASG-ControlManager User's Guide*.

As an alternative to creating a single MP-AID environment, you can utilize the concatenated MP-AID facility and create one or more secondary MP-AIDs, which together with a single read/write primary MP-AID can be accessed on a read-only basis by other Manager Products users.

At the simplest level, you might decide to create two MP-AIDs, one to contain the ASG-supplied members and the other to contain your installation-created corporate and user members.

For full details of this facility refer to the *ASG-Manager Products System Administrator's Guide*.

In this and subsequent chapters of this publication the creation and usage of a single MP-AID environment is assumed.

Whether a single or multiple MP-AID environment is created the basic requirement is to load up and make available for use the ASG-supplied MP-AID components.

In particular, it must be emphasized that it is essential for the successful execution of Manager Products to load up and make available at ALL times the contents of dataset MP.COM.UNLOAD. Failure to do so will make certain Manager Products functionality inoperable.

Creation of the MP-AID is considered to be the Systems Administrator's responsibility and is achieved, therefore, using the Systems Administrator's private command MP-AID CREATE. Command specifications for Systems Administrators' commands and a description of the MP-AID are given in the *ASG-Manager Products System Administrator's Guide*.

Once the MP-AID has been created, the following ASG-supplied datasets should be loaded onto the MP-AID from the installation tape, using the Systems Administrator's MP-AID LOAD command:

- MP.COM.UNLOAD, which contains ASG-supplied COMMAND members, Corporate Executive Routines, and UDS tables DU016 and DU777. Generally, the supplied members have names that begin with the characters MC, MP or £P. Wherever possible you should avoid using these characters when naming members on the MP-AID.
- MP.INFO.UNLOAD, which contains INFOBANK members. It should be noted that loading InfoBank is much faster if there are no existing InfoBank panels present on the MP-AID.

If you have the Corporate Dictionary/Repository Definition Export for IDD facility (selectable unit DYR-TE08), a set of pre-defined Translation Rules are also supplied on dataset MP.INFO.UNLOAD. These should also be loaded using the MP-AID LOAD command.

Having loaded all the required datasets, it is advisable to unload and then reload the MP-AID, in order to optimize directory placement and thus obtain improved response times when accessing the MP-AID. Details of unloading and reloading the MP-AID are provided in the *ASG-Manager Products System Administrator's Guide*.

Performance can be enhanced by increasing the size of the MP-AID buffer pool during load operations. ASG recommends a minimum of 30 buffers.

Step 4A - Create a BDAM-Organized MP-AID and Load Dataset MP.INFO.UNLOAD

To create the MP-AID you must execute Manager Products, having first configured your virtual machine using these commands:

```
DEFINE STORAGE nnnnK
IPL CMS
```

where:

nnnn is the storage requirement in kilobytes. This value is dependent on the installation option selected, the Manager Products present, the degree of Manager Products tailoring, and the commands to be utilized. Typical program sizes for the non-tailored versions of Manager Products are shown in Appendix E,"Virtual Storage Requirements for Running Manager Products," on page 157. It is expected that a typical total storage requirement would be 1,000K in excess of the values given in Appendix E, for the program code only.

You should then set up and execute a CMS EXEC containing the commands given in Figure 6,

where:

ffff is the user ID whose directory entry defines the mini-disk containing the Manager Products software.

gggg is the user ID for the user whose directory entry defines the mini-disk which will contain the BDAM MP-AID.

vaddr1 is the channel and unit number of the mini-disk containing the Manager Products software.

vaddr2 is the channel and unit number, to be used by the current user, for referencing the mini-disk containing the Manager Products software.

vaddr3 is the channel and unit number of the mini-disk which will contain the BDAM MP-AID.

vaddr4 is the channel and unit number, to be used by the current user, for referencing the mini-disk which will contain the MP-AID.

w is the filemode letter identifying the disk on which the Manager Products software is stored.

x is the filemode letter identifying the disk on which the MP-AID is to be created.

fn is the filename of the MP-AID to be created.

ft is the filetype of the MP-AID to be created.

nnnnn is the number of physical records allocated to the file.

Figure 6. Create a BDAM MP-AID and Load Dataset MP.INFO.UNLOAD

```
/*   MPX01 - Create and load a MANAGER Products BDAM MP-AID   */
/*                                                            */
'EXECOS'
'GLOBAL LOADLIB MPRLIB'
'CP LINK  ffff vaddr1 vaddr2  R'
'CP LINK  gggg vaddr3 vaddr4 MW'
'ACCESS vaddr2 w'
'ACCESS vaddr4 x'
'SET LDRTBLS 10'
'FILEDEF MPIN TERMINAL'
'FILEDEF MPOUT TERMINAL'
'FILEDEF MPRDIAG DISK MPRDIAG LISTING A1'
'FILEDEF MPAID DISK fn ft x6 (XTENT nnnnn'
'FILEDEF MPAIDR TAP1 SL 17 (BLOCK 9442'
/* Option-specific statements                               */
/* Option 1                                                 */
'OSRUN MPR00 PARM=LINE'
/* Option 2                                                 */
'LOADMOD CM02'
'START * LINE'
/* Option 3                                                 */
'LOADMOD MPR00'
'START * LINE'
/* End of option-specific statements                        */
'GLOBAL LOADLIB'
'RELEASE w (DET'
'RELEASE x (DET'
```

Until the MP-AID has been created, you cannot run ControlManager in full screen mode. The Manager Products commands to create and load the MP-AID must therefore be entered in line mode. These commands are as follows:

```
MP-AID CREATE ADMINISTRATOR logon-id PASSWORD password
LOGICAL-BLOCKSIZE llll PHYSICAL-BLOCKSIZE pppp ;
LOGON logon-id PASSWORD password ;
MP-AID BUFFERS nn;
MP-AID LOAD INFOBANK <TRANSLATION-RULES> ;
MP-AID STATUS ;
LOGOFF ;
```

where:

llll is the required logical blocksize for the MP-AID.

pppp is the required physical blocksize for the MP-AID.

nn is the number of MP-AID buffers to be allocated.

Step 4B - Load the MP-AID With Dataset MP.COM.UNLOAD

In addition to dataset MP.INFO.UNLOAD, you must also load dataset MP.COM.UNLOAD, which contains the ASG-supplied COMMAND members, Corporate Executive Routines, and UDS tables DU016 and DU777.

To load this dataset onto a BDAM-organized MP-AID you must execute Manager Products, having first configured your virtual machine using these commands:

```
DEFINE STORAGE nnnnK
IPL CMS
```

where *nnnn* is the storage requirement in kilobytes. This value is dependent on the installation option selected, the Manager Products present, the degree of Manager Products tailoring, and the commands to be utilized. Typical program sizes for the non-tailored versions of Manager Products are shown in Appendix E,"Virtual Storage Requirements for Running Manager Products," on page 157. It is expected that a typical total storage requirement would be 1,000K in excess of the values given in Appendix E, for the program code only.

You should then set up and execute a CMS EXEC containing the commands given in Figure 7,

where:

ffff is the user ID whose directory entry defines the mini-disk containing the Manager Products software.

gggg is the user ID for the user whose directory entry defines the mini-disk containing the BDAM MP-AID.

vaddr1 is the channel and unit number of the mini-disk containing the Manager Products software.

vaddr2 is the channel and unit number, to be used by the current user, for referencing the mini-disk containing the Manager Products software.

vaddr3 is the channel and unit number of the mini-disk containing the BDAM MP-AID

vaddr4 is the channel and unit number, to be used by the current user, for referencing the mini-disk containing the MP-AID.

w is the filemode letter identifying the disk on which the Manager Products software is stored.

x is the filemode letter identifying the disk on which the MP-AID is located.

fn is the filename of the MP-AID.

ft is the filetype of the MP-AID.

nnnnn is the number of physical records allocated to the file.

Figure 7. Load Dataset MP.COM.UNLOAD to a BDAM MP-AID

```
/*   MPX02 - Load the MP-AID with data set MP.COM.UNLOAD   */
/*                                                         */
'EXECOS'
'GLOBAL LOADLIB MPRLIB'
'CP LINK  ffff vaddr1 vaddr2 R'
'CP LINK  gggg vaddr3 vaddr4 MW'
'ACCESS vaddr2 w'
'ACCESS vaddr4 x'
'SET LDRTBLS 10'
'FILEDEF MPIN TERMINAL'
'FILEDEF MPOUT TERMINAL'
'FILEDEF MPRDIAG DISK MPRDIAG LISTING A1'
'FILEDEF MPAID DISK fn ft x6 (XTENT nnnnn'
'FILEDEF MPAIDR TAP1 SL 16 (BLOCK 9442'
/* Option-specific statements                             */
/* Option 1                                               */
'OSRUN MPR00'
/* Option 2                                               */
'LOADMOD CM02'
'START *'
/* Option 3                                               */
'LOADMOD MPR00'
'START *'
/* End of option-specific statements                      */
'GLOBAL LOADLIB'
'RELEASE w (DET'
'RELEASE x (DET'
```

When ControlManager starts executing you will be presented with a logon panel in which you should enter the same logon ID and password as was specified with the MP-AID CREATE command when the MP-AID was created, see "Step 4A - Create a BDAM-Organized MP-AID and Load Dataset MP.INFO.UNLOAD" on page 76.

Once you have successfully logged on you should enter the following Manager Products commands:

```
MP-AID BUFFERS nn;
MP-AID LOAD ALL ;
MP-AID STATUS ;
LOGOFF ;
```

where *nn* is the number of MP-AID buffers to be allocated.

7 Step 5 - Set Up Dictionaries

Introduction

Creating a dictionary is considered to be the responsibility of the person who will control that dictionary; that is, the dictionary Controller. Dictionary creation is achieved, therefore, using the dictionary Controller's CREATE command. A description of the CREATE command and its usage is given in the *ASG-Manager Products Controller's Manual*. Manager Products supports BDAM and VSAM-organized dictionaries.

When you create a new dictionary you may, if you wish, replace the default UDS table (DU001) with UDS table DU016, which is constructed from the MethodManager Information Engineering Knowledge base (MIEKB) Structure. UDS table DU016 is supplied in dataset MP.COM.UNLOAD, which will have been loaded from the installation tape when setting up the MP-AID. To use this UDS table on the dictionary, you should issue a CONTROL UDS command. For further details, refer to the *ASG-MethodManager Dictionary/Repository Information Model*.

The dictionary definitions for this Information Model are provided as part of the Manager Products UDS dictionary supplied as dataset MP.UDS.

The ASG-supplied dictionaries are set up by creating a dictionary and restoring the required ASG-supplied datasets into it using the dictionary Controller's RESTORE command.

Creating Dictionaries

Create a BDAM Dictionary

To create a dictionary you must execute Manager Products, having first configured your virtual machine using these commands:

```
DEFINE STORAGE nnnnK
IPL CMS
```

where *nnnn* is the storage requirement in K bytes. This value is dependent on the installation option selected, the Manager Products present, the degree of Manager Products tailoring, and the commands to be utilized. Typical program sizes for the non-tailored versions of Manager Products are shown in Appendix E,"Virtual Storage Requirements for Running Manager Products," on page 157. It is expected that a typical total storage requirement would be 1,000K in excess of the values given in Appendix E, for the program code only.

You should then set up and execute a CMS EXEC containing the commands given in Figure 8 on page 83, where:

ffff is the user ID whose directory entry defines the mini-disk containing the Manager Products software.

gggg is the user ID for the user whose directory entry defines the mini-disk which contains the BDAM MP-AID.

hhhh is the user ID whose directory entry defines the mini-disk which will contain the BDAM dictionary.

vaddr1 is the channel and unit number of the mini-disk containing the Manager Products software.

vaddr2 is the channel and unit number, to be used by the current user, for referencing the mini-disk containing the Manager Products software.

vaddr3 is the channel and unit number of the mini-disk which contains the BDAM MP-AID.

vaddr4 is the channel and unit number, to be used by the current user, for referencing the mini-disk which contains the MP-AID.

vaddr5 is the channel and unit number of the mini-disk which will contain the BDAM dictionary.

vaddr6 is the channel and unit number to be used by the current user for referencing the mini-disk which will contain the BDAM dictionary.

w is the filemode letter identifying the disk on which the Manager Products software is stored.

x is the filemode letter identifying the disk which contains the MP-AID.

y is the filemode letter identifying the disk on which the BDAM dictionary is to be created.

fn is the filename of the MP-AID or dictionary dataset.

ft is the filetype of the MP-AID or dictionary dataset.

dict is the name of the dictionary to be created.

nnnnn is the number of physical records allocated to the file.

Figure 8. Create a Manager Products BDAM Dictionary

```
/*   MPX03 Create a MANAGER Products BDAM Dictionary   */
/*                                                     */
'EXECOS'
'GLOBAL LOADLIB MPRLIB'
'CP LINK ffff vaddr1 vaddr2 R'
'CP LINK gggg vaddr3 vaddr4 MW'
'CP LINK hhhh vaddr5 vaddr6 MW'
'ACCESS vaddr2 w'
'ACCESS vaddr4 x'
'ACCESS vaddr6 y'
'SET LDRTBLS 10'
'FILEDEF MPIN TERMINAL'
'FILEDEF MPOUT TERMINAL'
'FILEDEF MPRDIAG DISK MPRDIAG LISTING A1'
'FILEDEF MPAID DISK fn ft x6 (XTENT nnnnn'
'FILEDEF dict DISK fn ft y6 (XTENT nnnnn'
'FILEDEF dictD DISK fn ft y6 (XTENT nnnnn'
'FILEDEF dictS DISK fn ft y6 (XTENT nnnnn'
'FILEDEF dictE DISK fn ft y6 (XTENT nnnnn'
'FILEDEF dictJ DISK fn ft y6 (XTENT nnnnn'
/* Option-specific statements                         */
/* Option 1                                           */
'OSRUN MPR00'
/* Option 2                                           */
'LOADMOD CM02'
'START *'
/* Option 3                                           */
'LOADMOD MPR00'
'START *'
/* End of option-specific statements                  */
'GLOBAL LOADLIB'
'RELEASE w (DET'
'RELEASE x (DET'
'RELEASE y (DET'
```

When ControlManager starts executing you will be presented with a logon panel in which you should enter the same logon-ID and password as was specified with the MP-AID CREATE command when the MP-AID was created, see Chapter 6, "Step 4 - Create and Load the MP-AID," on page 75.

Once you have successfully logged on you should enter these Manager Products commands:

```
CREATE dict MASTER authority ILB llll SLB llll DLB llll
IPB pppp SPB pppp DPB pppp RPB pppp LPB pppp WITH n STATUSES AND LOG UPDATES;
DICTIONARY dict;
AUTHORITY authority ;
QUERY DICTIONARY ;
LOGOFF ;
```

where:

llll is the required logical blocksize for the dictionary dataset

pppp is the required physical blocksize for the dictionary dataset.

The physical blocksize specified for the Recovery dataset must be at least as large as the largest of the logical blocksizes for the Source, Index, and Data Entries datasets.

The number of blocks needed for the Recovery dataset can be ascertained by monitoring recovery usage using the QUERY DICTIONARY command. The Recovery dataset will need to be considerably larger than for normal processing if you wish to run Logical Units of Work (LUWs) with ROLLBACK specified, since it must then be large enough to hold pre-update images of each update to be performed within the LUW. For details of LUWs, refer to panel MPRBC0060 in InfoBank.

Guidelines for dataset blocksize and buffer pool allocations are provided in the *ASG-Manager Products Performance Tuning* guide.

Set Up Manager Products Administration Dictionary

This section should be read in conjunction with the relevant section in the *ASG-Manager Products Controller's Manual*.

As stated in Chapter 1, "Overview of Installation," on page 11, ASG recommends that each installation set up its own Manager Products Administration Dictionary. The content of this dictionary will vary depending on the selectable units purchased and the needs of your organization. However, these are the steps to follow in setting up a Manager Products Administration Dictionary:

1. Create an empty dictionary as described in section "Creating Dictionaries" on page 82.

2. Restore the UDS dictionary (only applicable if you have the User Defined Syntax facility, selectable unit CMR-UD1). The UDS dictionary must be the first dataset to be restored.

3. Restore the ASG-supplied Corporate Executive Routines (necessary for tailoring or maintaining the Corporate Executive Routines supplied in dataset MP.COM.UNLOAD)

4. Restore the ASG-supplied InfoDictionary (only applicable if you have the Control Manager User Defined InfoSystem facility, selectable unit CMR-UD10)

5. Restore the ASG-supplied Translation Rules (only applicable if you have the Corporate Dictionary/ Repository Definition Export for IDD facility, selectable unit DYR-TE08)

6. Begin to set up the dictionary members (for example, PROFILEs) that when constructed into the MP-AID will control the environment in which your users will work. Details of how to construct members onto the MP-AID from the Manager Products Administration Dictionary are given in the *ASG-Manager Products Systems Administrator's Guide*.

The above procedure outlines how to set up the single Manager Products Administration Dictionary recommended by ASG. However, certain installations may need to keep their User Defined Syntax definitions and/or INFOBANK-PANEL members and/or TRANSLATION-RULE members in separate dictionaries. You can do this by creating separate dictionaries and restoring the appropriate dataset into the required dictionary.

Restore ASG-supplied UDS Dictionary

If you have purchased the ControlManager User Defined Syntax facility (selectable unit CMR-UD1), restore the ASG-supplied UDS dictionary from the installation tape into the empty dictionary by using the dictionary Controller's RESTORE ALL command. The job control requirements for achieving this are those job control statements specified as being compulsory for all users in Figure 9 on page 107 plus this statement:

```
FILEDEF dictR TAP1 SL n (BLOCK 9442
```

where:

dict is the name of the Manager Products Administration Dictionary.

n is the position of the MP.UDS dataset on your ASG-supplied release tape; this is given in the list of datasets provided with this tape.

Restore ASG-supplied Executive Routines

You need to restore the ASG-supplied Executive Routines if you wish to tailor any of them, or if maintenance of the Executive Routines becomes necessary (since maintenance of these members is carried out in the dictionary, not on the MP-AID).

Executive Routines are restored from the installation tape into the Manager Products Administration Dictionary using the dictionary Controller's RESTORE SOURCE command. The job control requirements for achieving this are those job control statements which are specified as being compulsory for all users in Figure 9 on page 107 plus this statement:

```
FILEDEF dictR TAP1 SL n (BLOCK 9442
```

where:

dict is the name of the Manager Products Administration Dictionary.

n is the position of the MP.CORP dataset on your ASG-supplied release tape; this is given in the list of datasets provided with this tape.

Dataset MP.CORP contains a SAVE SOURCE of the ASG-supplied Executive Routines used with these facilities:

- Corporate Dictionary/Repository Definition Import from DB2 (selectable unit DYR-TI12)
- Corporate Dictionary/Repository Definition Export to DB2 (selectable unit DYR-TE12)
- Corporate Dictionary/Repository Definition Import from SQL/DS (selectable unit DYR-TI32)
- Corporate Dictionary/Repository Definition Export to SQL/DS (selectable unit DYR-TE32)
- Workstation Interface (selectable unit CMR-WS01). The Executive Routines pertain to the Repository Diagram Generation feature.
- ADW/IEW Integration Facility (selectable units TE14, TE15, TI14 and TI15).

When the ASG-supplied Executive Routines have been restored into the Manager Products Administration Dictionary, only the source of these Executive Routine members will be available.

Restore ASG-supplied InfoDictionary

If you have purchased the ControlManager User Defined InfoSystem facility (selectable unit CMR-UD10), restore the ASG-supplied InfoDictionary from the installation tape into the Manager Products Administration Dictionary by using the Dictionary Controller's RESTORE SOURCE command. The job control requirements for achieving this are those job control statements which are specified as being compulsory for all users in Figure 9 on page 107 plus this statement:

```
FILEDEF dictR TAP1 SL n (BLOCK 9442
```

where:

dict is the name of the Manager Products Administration Dictionary.

n is the position of the MP.INFO dataset on your ASG-supplied release tape; this is given in the list of datasets provided with this tape.

When the InfoDictionary has been restored into the Manager Products Administration Dictionary, the source only of the ASG-supplied INFOBANK-PANEL members will be available.

Restore ASG-supplied Translation Rules

If you have purchased the DictionaryManager Corporate Dictionary/Repository Definition Export for IDD facility (selectable unit DYR-TE08), you should restore the ASG-supplied translation rules from the installation tape into the Manager Products Administration Dictionary by using the dictionary Controller's RESTORE SOURCE command. The job control requirements for achieving this are those job control statements which are specified as being compulsory for all users in Figure 9 on page 107, plus this statement:

```
FILEDEF dictR TAP1 SL n (BLOCK 9442
```

where:

dict is the name of the Manager Products Administration Dictionary.

n is the position of the MP.DYR.RULES dataset on your ASG-supplied release tape; this is given in the list of datasets provided with this tape.

When the ASG-supplied Translation Rules have been restored the Manager Products Administration Dictionary, the source only of the ASG-supplied TRANSLATION-RULE members will be available.

Set Up the Manager Products DEMO Dictionary

This section should be read in conjunction with the relevant section in the *ASG-Manager Products Controller's Manual*.

Your Manager Products installation tape contains a dataset, MP.DEMO. This example dictionary can be retrieved and set up as described in the *ASG-Manager Products Controller's Manual*. This requires a ControlManager run to create an empty dictionary and to restore the DEMO Dictionary. The job control statements required for restoring the DEMO dictionary are those job control statements which are specified as being compulsory for all users in Figure 9 on page 107, plus this statement:

```
FILEDEF dictR TAP1 SL n (BLOCK 9442
```

where:

dict is the name of the empty dictionary that will become the Manager Products DEMO Dictionary.

n is the position of the MP.DEMO dataset on your ASG-supplied release tape; this is given in the list of datasets provided with this tape.

Set Up the User Interface (POST) Dictionary

This section should be read in conjunction with the relevant section in the *ASG-Manager Products Controller's Manual*. If you have purchased the User Interface Facility (selectable unit CMR-UI1) and/or the User Defined Output facility (selectable unit DYR-UD15), a dataset MP.UIDICT is included on the installation tape. This dataset contains a dictionary of the User Interface output record formats and Access Call control parameter area definitions, and information relating to User Defined Output parameter numbers.

If you wish to make use of the POST dictionary, then it can be retrieved and set up as described in the *ASG-Manager Products Controller's Manual*.The job control requirements for restoring the POST dictionary are those job control statements which are specified as being compulsory for all users in Figure 9 on page 107, plus this statement:

```
FILEDEF dictR TAP1 SL n (BLOCK 9442
```

where:

dict is the name of the empty dictionary that will become the Manager Products User Interface Dictionary.

n is the position of the MP.UIDICT dataset on your ASG-supplied release tape; this is given in the list of datasets provided with this tape.

8 Step 6 - Satisfy Concurrent Usage Requirements

Introduction

When executing Manager Products under CMS, use of the ENQ and DEQ macros are not supported by CMS as part of the OS environment simulation. Hence there is no automatic protection against concurrent usage of any given dictionary or MP-AID. If such concurrency occurs, it is likely to corrupt the dictionary or MP-AID. You can eliminate (to a certain extent) the danger of corruption occurring by careful use and control of access to mini-disks containing dictionaries or an MP-AID. However, this is not the most efficient use of Manager Products where updates and interrogations can be required on a frequent basis.

Within Manager Products it is possible to establish an environment where concurrent access can occur without the risk of corruption and which is transparent to all users.

A number of control disks must be created, one for the MP-AID and two for each dictionary requiring concurrent usage. The Manager Products software issues internal CP LINK commands to these control disks in order to serialize access and updating of a dictionary or MP-AID.

Concurrent MP-AID Usage

Implementation

The following steps are required for implementation of the Manager Products concurrent usage capability for the MP-AID.

1. Define a new CMS user ID in the CP directory with two mini-disks allocated to it. The first mini-disk will contain only the shared MP-AID. The second mini-disk is required as a control disk through which all requests for MP-AID access are channeled. The requests are channeled via CP LINK commands issued by the Manager Products software. The space allocated to the control disk should be the minimum permitted.

   This is an example of the required CP directory entry:

```
MDISK 191 3350 100 030 PACK01 MR ALL ALL ALL
MDISK 192 3350 170 001 PACK01 MR ALL ALL ALL
```

2. Issue a CMS FORMAT command to format the mini-disk which will be used to hold the MP-AID. Any permissible blocksize can be specified with the CMS FORMAT command. If, however the blocksize of the mini-disk is not 800, then the MP-AID must be created and subsequently accessed with a file mode of 6 (update in place).

   The MP-AID dataset must be subsequently created with a physical blocksize which is equal to or a multiple of the blocksize allocated for the MP-AID mini-disk. ASG recommends that you format the CMS mini-disk to 4,096 byte blocks and the MP-AID to 8,192 byte physical blocks.

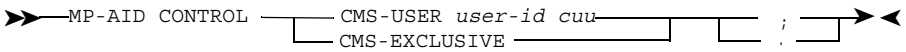
3. Create the MP-AID in accordance with the formatting rules of step 2.

4. Ensure that, for each user requiring access to the MP-AID, the relevant execution procedure includes a multiple write (MW) CP LINK command to the disk that contains the shared MP-AID. That is, MW must be specified as the access mode in the command. It is important to note that the execution procedure must not include a link to the control disk allocated to the MP-AID.

5. Initiate concurrent usage protection for the MP-AID by issuing an appropriate MP-AID CONTROL CMS-USER command. This command can be entered only by the Systems Administrator and should be issued when no other user is accessing the MP-AID.

   To override the MP-AID CONTROL CMS-USER command, an MP-AID CONTROL CMS-EXCLUSIVE command must be entered, in which case no concurrent usage protection for the MP-AID is provided by Manager Products software.

   The syntax for the MP-AID CONTROL CMS command is:



where:

user-id is the CMS user ID which has been allocated mini-disks to hold the MP-AID and an MP-AID control disk, as described in step 1 above.

cuu is the address of the control disk consisting of a one-digit channel number and a two-digit unit number.

For example, to initiate concurrent usage protection with the user ID MDI001 and control disk 192, enter:

```
MP-AID CONTROL CMS-USER MDI001 192 ;
```

Effect of Concurrent Usage Protection at Execution Time - Standard Usage

When the Manager Products enqueueing capability is enabled, all commands are enqueued. This means that when the enqueue capability is utilized, the following conditions apply:

- If an update command is using the MP-AID, then no other command, whether an interrogation or an update, can use the MP-AID. Thus, any commands submitted subsequent to the update command are put into a wait state and are queued according to order of submission.
- If an interrogation command is using the MP-AID, then:
  - If the next command submitted is an interrogation command, then the interrogation command is allowed concurrent usage of the MP-AID. Any number of interrogation commands can be processed simultaneously.
  - If the next command submitted is an update command, the update command is put into a wait state. Any commands (whether interrogation or update) submitted after the update command are put into a wait state, and are queued behind the update command in order of submission.

Concurrent Dictionary Usage

Implementation

Follow these steps to implement the Manager Products concurrent usage capability for dictionaries:

1. Define a new CMS user ID in the CP directory with three mini-disks allocated to it. The first mini-disk will contain only the shared dictionary files. The second and third mini-disks are required as control disks through which all requests for dictionary access are channeled. The requests are channeled via CP LINK commands issued by the Manager Products software. One of these control disks is a serialization disk which is used to queue requests until the dictionary is available. The other is the actual dictionary utilization control disk which is used to determine when the next queued request for dictionary processing can be accepted. The space allocated to the control disks should be the minimum permitted.

   This is an example of the required CP directory entry:

```
MDISK 193 3350 100 070 PACK02 MR ALL ALL ALL
MDISK 194 3350 170 001 PACK02 MR ALL ALL ALL
MDISK 195 3350 171 001 PACK02 MR ALL ALL ALL
```

2. Issue a CMS FORMAT command to format the mini-disk which will be used to hold the dictionary. Any permissible blocksize can be specified with the CMS FORMAT command. If, however the blocksize of the mini-disk holding the dictionary is not 800 then the files constituting the dictionary must be created and subsequently accessed with a file mode of 6 (update in place).

   Each of the dictionary datasets/files must be created with a physical blocksize which is equal to or a multiple of the blocksize specified for the mini-disk.

   If, for example, an FBA type of disk is used, which requires that the blocksize be a multiple of 512, the following physical blocksizes could be specified:

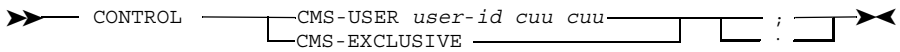
   - 4,096 for the mini-disk
   - 8,192 for the Index and Data Entries datasets
   - 4,096 for the Source dataset
   - 4,096 for the Recovery dataset
   - 4,096 for the Log dataset, if logging is applied to the dictionary.

3. Create the dictionary in accordance with the formatting rules of step 2.

4. Ensure that, for each user requiring access to the dictionary, the relevant execution procedure includes a multiple write (MW) CP LINK command to the disk that contains the shared dictionary. That is, MW must be specified as the access mode in the command. It is important to note that the execution procedure must not include a link to either of the control disks allocated to the dictionary.

5. Initiate concurrent usage protection for the dictionary by issuing an appropriate CONTROL CMS-USER command. This command can be entered only by the dictionary Controller and should be issued when no other user is accessing the dictionary.

   To override the CONTROL CMS-USER command, a CONTROL CMS-EXCLUSIVE command must be entered, in which case no concurrent usage protection for the dictionary is provided by the Manager Products software. The syntax for the CONTROL CMS command is:



   Where:

   user-id is the CMS user ID which has been allocated mini-disks to hold the dictionary files and dictionary control disks, as described in step 1 above.

   cuu is the address of a control disk consisting of a one-digit channel number and a two-digit unit number.

For example, to initiate concurrent access protection with the user ID MDI001 and control disks 194 and 195, enter:

```
CONTROL CMS-USER MDI001 194 195 ;
```

Disk 194 then will be used as the serialization control disk and 195 as the dictionary utilization control disk.

Effect of Concurrent Usage Protection at Execution Time - Standard Usage

When the Manager Products enqueueing capability is enabled, all commands are ENQueued. This means that, when the enqueue capability is utilized, the following conditions apply:

- If an update command is using the dictionary, then no other command, whether an interrogation or an update, can use the dictionary concurrently. Thus, any commands submitted subsequent to the update command are put into a wait state and are queued according to order of submission.
- If an interrogation command is using the dictionary, then:
  - If the next command submitted is an interrogation command, then the interrogation command is allowed concurrent usage of the dictionary. Any number of interrogation commands can be processed simultaneously.
  - If the next command submitted is an update command, then the update command is put into a wait state. Any commands, whether interrogation or update, that are submitted after the update command are put into a wait state, and are queued behind the update command in order of submission.

Effect of Concurrent Usage Protection at Execution Time - Alternative Usage

The Systems Administrator's command, SET INTERROGATE-ENQ, (see the *ASG-Manager Products Systems Administrator's Guide*), provides an alternative enqueueing procedure whereby normal enqueues used with interrogation commands are released to allow updates to take place. This can greatly improve response times for all users.

When the alternative enqueue mode is used the following conditions apply:

- If an update command is using the dictionary, no other command, whether an interrogation or an update, can access the dictionary at the same time. Thus, any commands submitted after the update command are put into a wait state and are queued in order of submission.
- If only interrogation commands are using the dictionary, one update command is allowed concurrent usage of the dictionary. However, as soon as an update command gains access to the dictionary, no subsequent commands, whether update or interrogation, can gain access until the update has ended. Subsequent commands are put into a wait state as described above.

Thus, if the alternative enqueue mode is in operation, it is possible for one or more interrogation commands and one update command to be processed simultaneously. This means that update commands do not have to wait until long interrogations are completed. However, certain commands cannot be processed simultaneously with other Manager Products commands. See *ASG-Manager Products Systems Administrator's Guide* for details.

If an update command starts while an interrogation command is in progress, it is possible for the interrogation command to access an area of the dictionary that has been updated. Thus, information may be read that was not current at the start of the interrogation command. When this situation occurs, the interrogation command is suspended and an 'I/O flush' takes place. A diagnostic message is issued each time an I/O flush occurs.

An I/O flush comprises the following sequence of actions:

- A shared enqueue is issued, to ensure that the update command has completed.
- If the update command failed to complete, the dictionary is recovered to the state it was in before the update command began.
- The buffer pools for the Index, Source, and Data Entries datasets are flushed.
- The shared enqueue is released.
- The interrogation command is reactivated from the point at which it was interrupted.

There is no limit to the number of I/O flushes that can be performed, unless a limit is specifically set by the Systems Administrator. The I/O flush limit is set using the Systems Administrator's command, SET IO-FLUSH-LIMIT.

If you do set a limit, an interrogation command is terminated once the limit is reached. You could, therefore, prevent an interrogation command from accessing data that has been updated while the interrogation command was active, by setting the I/O flush limit to one. In this case, any interrogation command is terminated at the first occurrence of an I/O flush.

Checklist for a Successful Implementation of Concurrent Usage

In a shared environment where concurrent usage of the MP-AID and dictionaries is required you should check that:

- The MP-AID and dictionary physical blocksizes are equal to or a multiple of the CMS mini-disk blocksize
- Filemode 6 is used in the FILEDEF statements when running Manager Products if the CMS mini-disk blocksize is not 800 bytes
- The MP-AID CONTROL CMS-USER command has been correctly applied by issuing an MP-AID STATUS command
- The CONTROL CMS-USER command has been correctly applied by issuing a QUERY DICTIONARY command
- The CMS user IDs allocated with those mini-disks which hold the MP-AID/dictionary files and control disks are established with a CP DIRECTORY entry containing the NOLOG keyword
- No virtual disks are attached to your virtual machine with addresses of 5FD, 5FE, or 5FF. This is because the Manager Products enqueueing software utilizes these addresses when linking to the appropriate control disks
- There are no links established at any time to any of the Manager Products control disks
- The control disks are exclusively used for a given dictionary or MP-AID to be shared. If this rule is not observed, deadlock can occur.

9 Step 7 - Linking External Software To Manager Products Software

Introduction

There are currently three modes of communication between Manager Products software and external software:

- Access Call, which allows you to run Manager Products from a user program using Access Call, available with the User Interface facility (CMR-UI1). See "Making Access Call Programs Available to Manager Products" on page 95 for further details.
- The User Defined Function facility which allows User Defined Functions to be written in languages such as BAL, COBOL, or PL/I and referenced within an Executive Routine. This is available with the User Defined Commands (CMR-UD05) or Translation and Transfer Engine (DYR-TE00) facilities. See "Making User-defined Functions Available to Manager Products" on page 97 for further details.
- The SQL Import facility, which allows you to access SQL environments. This is available with the Corporate Dictionary/Repository Definition Import from SQL facility (selectable unit DYR-TI32).

Details of how to link external software to Manager Products are provided in the these sections.

Making Access Call Programs Available to Manager Products

ASG assumes that the required Access Call programs have been compiled and the object modules (text files) have been generated and are available for loading. The instructions necessary to make Access Call programs available to Manager Products are dependent upon which Manager Products execution option was selected in Step 2. See Chapter 4, "Step 2 - Select and Generate an Executable Version of Manager Products," on page 31.

Details of executing Manager Products (with a BDAM dictionary and BDAM MP-AID) for an Access Call program can be found in "Executing an Access Call Program:" on page 107.

Setting up Access Call Programs for use with Manager Products - Execution Option 1

Enter these CMS commands:

```
LOAD userprog2 (MAP ORIGIN 20000
GENMOD userprog1 (FROM userprog2
```

where:

userprog2 is the name of the required text file.

userprog1 is the name of the user program to be executed.

If the entry point of *userprog2* cannot be determined by the CMS Loader, the LOAD statement should be:

```
LOAD userprog2 (RESET entrypoint MAP ORIGIN 20000
```

If the ASG-supplied User Interface Module (DMRUS) is to be dynamically loaded at execution time, then DMRUS may be excluded from the GENMOD module by using the NOAUTO option with the LOAD command.

Setting up Access Call Programs for use with Manager Products- Execution Option 2

Enter these CMS commands:

```
LOAD userprog2 (MAP ORIGIN 21000
GENMOD userprog1 (FROM userprog2
```

where:

userprog2 is the name of the required text file.

userprog1 is the name of the user program to be executed.

If the entry point of *userprog2* cannot be determined by the CMS Loader, the LOAD statement should be:

```
LOAD userprog2 (RESET entrypoint MAP ORIGIN 21000
```

If the ASG-supplied User Interface Module (DMRUS) is to be dynamically loaded at execution time, then DMRUS may be excluded from the GENMOD module by using the NOAUTO option with the LOAD command.

Setting up Access Call Programs for use with Manager Products - Execution Option 3

Enter these CMS commands:

```
LOAD userprog2 (MAP ORIGIN 1E0000
GENMOD userprog1 (FROM userprog2
```

where:

userprog2 is the name of the required text file.

userprog1 is the name of the user program to be executed.

If the entry point of *userprog2* cannot be determined by the CMS Loader, the LOAD statement should be:

```
LOAD userprog2 (RESET entrypoint MAP ORIGIN 1E0000
```

If the ASG-supplied User Interface Module (DMRUS) is to be dynamically loaded at execution time, then DMRUS may be excluded from the GENMOD module by using the NOAUTO option with the LOAD command.

Making User-defined Functions Available to Manager Products

If you have the User-Defined Commands (selectable unit CMR-UD05) or the Translation and Transfer Engine (selectable unit DYR-TE00) facilities, you may set up user-defined functions in any language capable of handling the argument table passed by the Manager Products Procedures Language. These functions may then be invoked in a Manager Products environment from within an Executive Routine.

Before you can invoke a user-defined function, you must:

- Specify the user-defined function by adding a function definition to the source module MPLUF
- Assemble the source module MPLUF.

Invoking User-defined Functions

Before you can invoke a user-defined function it must first be specified by adding a function definition to the source module MPLUF. MPLUF is supplied as a member of dataset MP.CMSRCE.

The name of each user-defined function is specified within MPLUF using the macro MPLFD (also supplied as a member of dataset MP.CMSRCE). For each user-defined function you need to specify an entry of the form:

```
MPLFD TYPE=ENTRY,NAME=function-name
```

where *function-name* is the name of a function referred to from within an Executive Routine.

When specifying a function, there is an optional additional operand, EP (Entry Point) you can use, which allows you to link the function to a module whose name differs from the function name. For example:

```
MPLFD TYPE=ENTRY,NAME=SORTR,EP=KLHJY42
```

The operands for NAME and EP can be up to 8 characters in length. You may add more function names to source module MPLUF at any time. Further entries are added to the table; they do not overwrite function names already present. Entries are deleted by removing the function name from the table.

Once you have defined all the user-defined functions you intend using, you must then assemble source module MPLUF in order to create a new text file (MPLUF). When you subsequently wish to invoke a function, the text file MPLUF is dynamically loaded, together with the text files representing the object code from the assembly or compilation of your user defined functions, and control is passed to the appropriate function. You must ensure that the required text files are available at execution time.

Once loaded, all functions are retained in virtual storage until the end of the Manager Products session.

You must assemble source module MPLUF each time the table of definitions is changed.

Refer to "Tailoring Source Modules" on page 70 for details of tailoring and assembling source modules.

Accessing SQL/DS Environments From Manager Products

The Corporate Dictionary/Repository Translation and Transfer Engine (selectable unit DYR-TE00) and the Dictionary/Repository Definition Import for SQL/DS facility (selectable unit DYR-TI32) enable you to use the DictionaryManager Dynamic SQL Services feature and the DictionaryManager EXTRACT SQL command respectively, when running Manager Products under CMS.

There are some additional installation steps to allow access to the SQL/DS environment during a Manager Products session. These are the steps required:

1. Load the Manager Products Access Module onto the SQL/DS database.

2. Link-edit the Manager Products SQL/DS components.

3. Define the required Manager Products Import views to SQL/DS.

4. Modify MP-AID EXECUTIVE member MPDY42DFLT.

There are no special job control requirements to enable access to the SQL environment while running Manager Products. However, you need to have DBA privileges to perform EXTRACT SQL.

It is assumed that you have already performed the following general Manager Products installation steps:

- Copied dataset MP.CMS to disk.
- Copied dataset MP.CMSRCE to disk to create a Manager Products MACLIB.
- Generated an executable version of the Manager Products software.
- Created an MP-AID and Administration dictionary.
- Loaded dataset MP.COM.UNLOAD onto the MP-AID.
- Restored dataset MP.CORP onto your Administration dictionary.

The installation instructions given are compatible for both SQL/DS versions 2.1 and 2.2.

It is assumed that users wishing to install and use the SQL/DS interface have the normal environment for SQL/DS access.

Step 1 - Load the Manager Products Access Module onto the SQL/DS database

The Manager Products SQL/DS Access Module (or PACKAGE) is supplied as member MPSQL of dataset MP.CMSRCE and must be loaded to your SQL/DS database using the SQL/DS DBSU utility.

These CMS commands can be used to achieve this:

```
FILEDEF INMOVE DISK maclib (MEMBER MPSQL
FILEDEF OUTMOVE DISK MPSQL PACKAGE A1
MOVEFILE

FILEDEF PKIN DISK MPSQL PACKAGE A1
EXEC SQLDBSU
RELOAD PACKAGE(MPSQL) REPLACE INFILE(PKIN)
```

where *maclib* is the name of the Manager Products MACLIB created by loading dataset MP.CMSRCE from the release tape.

Step 2 - Link-Edit the Manager Products SQL/DS Components

Execute the ASG-supplied EXEC MPX6 to link-edit the various components of the Manager Products SQL/DS Interface into the Manager Products CMS LOADLIB called MPRLIB.

You must ensure that you have access to the SQL/DS mini-disk containing the ARIRVSTC TEXT file as it is required as part of the above link-edit process.

Step 3 - Define the Required Manager Products SQL/DS Import Views to SQL/DS Using the SQLDBSU Command

You must create the required Manager Products views in SQL/DS using the SQLDBSU command.

ASG supplies two sets of views as members of dataset MP.CMSRCE. If you have SQL/DS version 2.1only, you should use member SQLVIEW1. If you have SQL/DS version 2.2, you should use member SQLVIEW2.

Having copied dataset MP.CMSRCE to disk and created a MACLIB, the appropriate member should be moved from the MACLIB to a CMS file. You can use the CMS MOVEFILE command to achieve this, as shown below.

```
FILEDEF INMOVE DISK maclib (MEMBER SQLVIEWx
FILEDEF OUTMOVE DISK fn ft fm
MOVEFILE
```

where:

maclib is the name of the Manager Products MACLIB created by loading dataset MP.CMSRCE from the release tape

x is either 1 or 2.

As supplied, the name of each view includes a prefix (MP_VIEW_), which is preceded by a variable, owner ID. Before creating your views, you need to replace the variable with an actual owner identification. ASG recommends that you use the CMS logon ID of your systems administrator. To insert an owner identification, you can use the XEDIT CHANGE command. For example:

```
CHANGE /OWNERID/SYSAD/ * *
```

To create the views you should enter this command:

```
EXEC SQLDBSU SYSIN(fn ft fm) SYSPRINT(TERM)
```

where *fn*, *ft* and *fm* represent the filename, filetype, and filemode respectively of the previously created CMS file.

Once SQLVIEW1 or SQLVIEW2 have been processed by SQLDBSU you should ensure that all returned SQLCODES are zero.

Step 4 - Modify MP-AID Executive Member MPDY42DFLT

A source version of the MP-AID EXECUTIVE member MPDY42DFLT is supplied as an EXECUTIVE-ROUTINE member in dataset MP.CORP.

Source member MPDY42DFLT contains a SET EXT_PREFIX statement in which you need to provide a value for the variable owner ID. You should insert the same owner identification as that designated at the time the views were created. As stated in the preceding section, ASG recommends that the owner identification should be the CMS logon ID of your systems administrator, for example:

```
SET EXT_PREFIX :SYSAD.MP_VIEW_:
```

Having provided a suitable owner ID, you should replace the ASG-supplied version of MPDY42DFLT on the MP-AID using the CONSTRUCT command.

10 Step 8 - Run Manager Products with BDAM MP-AID and BDAM Dictionaries

In order to successfully execute Manager Products you must ensure that your virtual machine is correctly configured using the following commands:

```
DEFINE STORAGE nnnnK
IPL CMS
```

Where *nnnn* is the storage requirements in kilobytes. This value is dependent on the installation option selected, the Manager Products present, the degree of Manager Products tailoring, and the commands to be utilized. Typical program sizes for the non-tailored versions of Manager Products are shown in Appendix E,"Virtual Storage Requirements for Running Manager Products," on page 157. It is expected that a typical total storage requirement would be 1,500K in excess of the values given in Appendix E, for the program code only.

You should then set up a CMS EXEC containing the commands given in Table 17 on page 102. These assumptions have been made:

- A BDAM MP-AID has already been created for use in a shared environment.
- A BDAM dictionary has already been created for use in a shared environment.

Note:

There are no special job control requirements to enable access to SQL/DS environments while running Manager Products. However, there are some additional installation steps you need to perform and you must have appropriate SQL/DS privileges in order to access and/or update SQL/DS data. Details of accessing SQL/DS environments from Manager Products are given in "Accessing SQL/DS Environments From Manager Products" on page 98.

Definitions of the variables used in Table 17 on page 102 are as follows:

ffff is the user ID whose directory entry defines the mini-disk containing the Manager Products software.

vaddr1 is the channel and unit number of the mini-disk containing the Manager Products software.

vaddr2 is the channel and unit number used by the current user for referencing the mini-disk containing the Manager Products software.

gggg is the user ID for the user whose directory entry defines the mini-disk containing the BDAM MP-AID.

vaddr3 is the channel and unit number of the mini-disk containing the BDAM MP-AID.

vaddr4 is the channel and unit number, to be used by the current user, for referencing the mini-disk containing the MP-AID.

hhhh is the user ID whose directory entry defines the mini-disk containing the BDAM dictionary.

vaddr5 is the channel and unit number of the mini-disk containing the BDAM dictionary.

Table 17. Running Manager Products Using a BDAM MP-AID and Dictionary

JCL Step	Requirement	Note
/* MPX04 Running MANAGER Products (BDAM MP-AID */		
/* and dictionary) */		
'EXECOS'	Always	
'GLOBAL LOADLIB MPRLIB'	Always	
'CP LINK *ffff vaddr1 vaddr2* R'	Always	
'CP LINK *gggg vaddr3 vaddr4* MW'	Always	
'CP LINK *hhhh vaddr5 vaddr6* MW'	Always	
'ACCESS *vaddr2 w*'	Always	
'ACCESS *vaddr4 x*'	Always	
'ACCESS *vaddr6 y*'	Always	
'SET LDRTBLS 10'	Always	
'FILEDEF MPIN TERMINAL'	Always (Primary input device)	
'FILEDEF MPOUT TERMINAL'	Always (Primary output device)	
'FILEDEF MPRDIAG DISK MPRDIAG LISTING A1'	Always (Formatted dump output)	
'FILEDEF MPAID DISK *fn ft x*6 (XTENT *nnnnn*'	Always (Enables use of MP-AID)	
'FILEDEF dict DISK *fn ft y*6 (XTENT *nnnnn*'	Any dictionary access	Note 1
'FILEDEF *dict*D DISK fn ft x6 (XTENT *nnnnn*'	Any dictionary access	Note 1
'FILEDEF *dict*S DISK *fn ft x*6 (XTENT *nnnnn*'	Any dictionary access	Note 1
'FILEDEF *dict*E DISK *fn ft x*6 (XTENT *nnnnn*'	Any dictionary access	Note 1
'FILEDEF *dictJ* DISK *fn ft x6* (XTENT *nnnnn*'	Any dictionary access (logging active)	Note 1
/* Optional statements for all users */		
'FILEDEF *ddname* DISK *fn ft fm* (XTENT *nnnnn*'	Concatenated MP-AID	Note 18
'FILEDEF *ddname* DISK *fn ft fm*'	DesignManager (output dataset for the PRODUCE command)	Note 2

Table 17. Running Manager Products Using a BDAM MP-AID and Dictionary

JCL Step	Requirement	Note
'FILEDEF *ddname* DISK *fn ft fm*'	DictionaryManager (output dataset for the TRANSFER command)	Note 3
'FILEDEF *ddname* DISK *fn ft fm*'	Alternative output dataset (for SWITCH OUTPUT command)	Note 4
'FILEDEF *ddname* DISK *fn ft fm*'	Automation of Set Up (DMR-AS1, DMR-AS2) input dataset	Note 5
'FILEDEF *ddname* DISK *fn ft fm*'	Source Language Generation (DMR-SL1 to DMR-SL9)	Note 6
'FILEDEF *ddname* DISK *fn ft fm*'	Procedures Language output dataset	Note 7
'FILEDEF OBJECT DISK *fn ft fm*'	ADW/IEW Import/Export dataset	Note 21
'FILEDEF ASSOC DISK *fn ft fm*'	ADW/IEW Import/Export dataset	Note 21
'FILEDEF TEXT DISK *fn ft fm*'	ADW/IEW Import/Export dataset	Note 21
'FILEDEF PROP DISK *fn ft fm*'	ADW/IEW Import/Export dataset	Note 21
'FILEDEF IEWXOI DISK *fn ft fm*'	ADW/IEW Import/Export dataset	Note 21
'FILEDEF IEWXAI DISK *fn ft fm*'	ADW/IEW Import/Export dataset	Note 21
'FILEDEF IEWXTI DISK *fn ft fm*'	ADW/IEW Import/Export dataset	Note 21
'FILEDEF IEWXPI DISK *fn ft fm*'	ADW/IEW Import/Export dataset	Note 21
'FILEDEF *ddname* DISK *fn ft fm*'	Generic Import input dataset	Note 20
'FILEDEF MPRPOST DISK *fn ft fm*'	User Interface (CMR-UI1) output POST dataset	Note 8
'FILEDEF MPRACWF DISK *fn ft fm*'	User Interface (CMR-UI1) Access Call Work File	Note
'FILEDEF *ddname* DISK *fn ft fm*'	Produces Language trace output (default ddname MPTRACE)	Note 21
'FILEDEF MPRECORD DISK *fn ft fm*'	REPLAY facility	Note 22
'FILEDEF MPRESULT DISK *fn ft fm*'	REPLAY facility	Note 22
'FILEDEF MPSPRINT DISK *fn ft fm*'	REPLAY facility	Note 22
/* Optional statements for Dictionary Controller */		
/* and Master Operators */		

Table 17. Running Manager Products Using a BDAM MP-AID and Dictionary

JCL Step	Requirement	Note
'FILEDEF *dict*V DISK *fn ft fm*'	SAVE/UNLOAD output dataset	Note 9
'FILEDEF *dict*R DISK *fn ft fm*'	RESTORE/RELOAD input dataset	Note 10
'FILEDEF *dict*G DISK *fn ft fm*'	RELOAD/ROLL-FORWARD archived input dataset	Note 11
'FILEDEF *dict*H DISK *fn ft fm*'	LOG ARCHIVE output dataset	Note 12 and Note 13
'FILEDEF *dict*G DISK *fn ft fm*'	LOG ARCHIVE input dataset	Note 12 and Note 14
'FILEDEF *dict*G DISK *fn ft fm*'	LOG ANALYSIS/AUDIT when input from archive dataset	Note 15
'FILEDEF *ddname* DISK *fn ft fm*'	AUDIT ONTO output dataset	Note 19
/* Optional statements for Systems Administrators */		
'FILEDEF MPAIDV DISK *fn ft fm*'	MP-AID UNLOAD output dataset	Note 16
'FILEDEF MPAIDR DISK *fn ft fm*'	MP-AID LOAD/MP-AID RELOAD input dataset	Note 17
/* Compulsory option-specific statements for all users */		
/* Option 1 */		
'OSRUN MPR00'		
/* Option 2 */		
'LOADMOD CM02'		
START *		
/* Option 3 */		
'LOADMOD MPR00'		
'START *'		
/*Compulsory statements for all users */		
'GLOBAL LOADLIB'		
'RELEASE *w* (DET'		
'RELEASE *x* (DET'		
'RELEASE *y* (DET'		

Where:

vaddr6 is the channel and unit number, to be used by the current user, for referencing the mini-disk containing the BDAM dictionary.

w is the filemode letter identifying the disk which contains the Manager Products software.

x is the filemode letter identifying the disk which contains the MP-AID.

y is the filemode letter identifying the disk which contains the BDAM dictionary.

fn is the filename of the MP-AID, dictionary or other partitioned/sequential dataset.

ft is the filetype of the MP-AID, dictionary or other partitioned/sequential dataset.

fm is the filemode of any partitioned/sequential dataset.

dict is the name of the dictionary.

nnnnn is the number of physical records allocated to a dictionary or MP-AID dataset.

ddname is as variously specified in "Notes" on page 105:

Notes

1. Dictionary Datasets:

   The dictionary comprises four or five datasets, each of which must be declared to the operating system.

   The ddname variable `dict` represents the name of the dictionary as known by users. The suffixes indicated for the Data Entries, Source, Index, and Recovery datasets are compulsory. The FILEDEF command for the Log dataset can be omitted from the execution EXEC if logging is not applied to the dictionary.

   Any other dictionaries accessed in the same job step must have additional FILEDEF commands for the four or five datasets constituting the dictionary.

2. DesignManager PRODUCE Dataset:

   This additional FILEDEF command is required to define a sequential source library dataset for output from the DESIGNMANAGER PRODUCE command.

   There is no default ddname. It must be specified via the ONTO clause of the PRODUCE command.

3. DictionaryManager Datasets:

   This additional FILEDEF command is required to define a partitioned or sequential source library dataset for output from the TRANSFER command, where ddname must be the same as that specified via the ONTO clause of the TRANSFER command.

   There is no default ddname for this dataset.

4. Alternative Output Dataset:

   If SWITCH OUTPUT commands are to be used during the Manager Products run to switch output to an alternative output dataset then the additional FILEDEF command is required to define the output dataset. Any valid name may be used as the ddname. The specified ddname is declared using the SWITCH OUTPUT command.

5. Automation of Set Up Datasets:

   Under CMS, input for the Automation of Set Up facility's CONVERT command must be from a partitioned dataset containing COBOL or PL/I source statements in card image format. A partitioned dataset can be created using the CMS MACLIB command. To run ControlManager when the Automation of Set Up command CONVERT is to be used, the additional FILEDEF command is required to define the input dataset, where ddname is the logical file name of the dataset from which the input source statements are to be read. It is the name that can be stated in the SYSNAME clause of the CONVERT command.

   If the SYSNAME clause is omitted from the CONVERT command, then the ddname assumed is:

   - COBLIB for COBOL conversions
   - PLILIB for PL/I conversions.

   Therefore, if the CONVERT command is to be used without the SYSNAME clause, the ddname specified must be either COBLIB or PLILIB as applicable.

   You do not need any additional FILEDEF commands for the Automation of Set Up MERGE command.

6. Source Language Generation Datasets:

   This additional FILEDEF command is required to define a partitioned or sequential source library dataset for output from the PRODUCE command.

   The ddname to be used for the output source library dataset can be declared as the value of the parameter DDNAME in the relevant installation macro. The supplied default value of this parameter is GENLIB. If this name is acceptable, the DDNAME parameter of the relevant macro need not be tailored, and GENLIB can be declared as the ddname in the FILEDEF command. Otherwise, the relevant installation macro should be tailored to ensure that the name declared as the value of the parameter DDNAME is the same as the name normally used in the FILEDEF command. The value declared or defaulted for the parameter DDNAME can, however, be overridden at run time by including an ONTO file-name clause in the PRODUCE command (see the Source Language Generation manual). Thus, the ddname used in the FILEDEF command must be:

   - The ddname declared in the ONTO clause of the PRODUCE command that initiates the source language generation, if the PRODUCE command includes an ONTO clause;
   - The name declared as the value of the parameter DDNAME when the relevant installation macro was tailored, if a value was declared for DDNAME;
   - GENLIB.

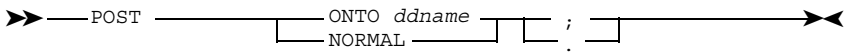
7. Procedures Language SENDF Output Dataset:

   This additional FILEDEF command is required when the Executive Command SENDF, specifying partitioned or sequential output, is to be used.

8. User Interface Optional Output POST Dataset:

   This additional FILEDEF command is required when POST or MAIL commands are to be issued.

   The default ddname established at logon to Manager Products is MPRPOST. This can be changed by use of the command:



9. Executing an Access Call Program:

   This additional FILEDEF command is required when the Access Call Work File is to be used. The ddname must be MPRACWF.

   In order to run the Access Call program, the option-specific statements given in Table 17 on page 102 should be replaced by those given in Figure 9 on page 107.

   Before executing an access call program it is necessary to link-edit your user program with the ASG-supplied access call interface module (DMRUS). This is described in Chapter 9, "Step 7 - Linking External Software To Manager Products Software," on page 95.

Figure 9. Option-specific Statements when Running Manager Products from a User Program

```
+
+
+
/* Option-specific requirements when executing an Access Call  */
/*                                                             */
/* Option 1                                                    */
'LOADMOD user-prog'
'START *'
/* Option 2                                                    */
'LOADMOD CM02'
'LOADMOD user-prog'
'START *'
/* Option 3                                                    */
'LOADMOD MPR00'
'LOADMOD user-prog'
'START *'
+
+
+
+
```

VM/SP6, VM/XA, and VM/ESA users must add the (PRES option to the `LOADMOD user-prog` statement in Option 2 and 3.

10. SAVE/UNLOAD Datasets:

    If a SAVE or UNLOAD command is to be issued in the course of a run, then this additional FILEDEF command is needed to define the output dataset. The specified ddname must be the name of the dictionary from which it is output, with the suffix V.

11. RESTORE/RELOAD Datasets:

    If a RESTORE or RELOAD command is to be issued in the course of a run, then this additional FILEDEF command is needed to define the input dataset. The ddname of the dataset input to a RESTORE or RELOAD command must be the name of the dictionary to which it is input, with the suffix R.

12. RELOAD and ROLL-FORWARD LOG-ARCHIVE Dataset:

    The RELOAD command can invoke roll-forward either automatically or by inclusion of the keyword ROLL-FORWARD. If roll-forward requires input from a log archive dataset, then this additional FILEDEF command is needed to define the input dataset.

    The ddname of the input log archive dataset must be the name of the dictionary with the suffix G.

13. LOG ARCHIVE General Information:

    The LOG ARCHIVE command archives the Log dataset to disk or tape. Log archiving works in a cyclical manner. A log archiving cycle commences whenever an UNLOAD command or a LOG BACKUP-DETAILS command is accepted. When the first LOG ARCHIVE command following one of these commands is processed, a new log archive dataset is created. Subsequent LOG ARCHIVE commands up to the time the next UNLOAD command or LOG BACKUP-DETAILS command is accepted require the dataset output from the previous log archiving run as input. This dataset is merged with transactions from the dictionary's Log dataset to produce the output log archive dataset.

14. LOG ARCHIVE Output Dataset:

    For any Manager Products run that includes the LOG ARCHIVE command, this additional FILEDEF command is required to define the output log archive dataset.

    The ddname of the output log archive dataset must be the name of the dictionary with the suffix H.

15. LOG ARCHIVE Optional Input Dataset:

    The ddname of any input log archive dataset must be the name of the dictionary with the suffix G. This additional FILEDEF command is required in respect of this dataset for every log archiving run except the first in each cycle.

16. LOG ANALYSIS and AUDIT Datasets:

    This additional FILEDEF command is required if a Manager Products run includes either:

    - A LOG ANALYSIS command,
    - An AUDIT command that is to operate on a archive dataset (If the ControlManager Audit and Security facility, selectable unit CMR-DD3, is installed).

    The ddname of the input log archive dataset must be the name of the dictionary with the suffix G. Concatenated input log archive datasets are not permissible.

17. MP-AID UNLOAD Datasets:

    If an MP-AID UNLOAD command is to be issued in the course of a run, then this additional FILEDEF command is needed to define the output dataset. The specified ddname must be MPAIDV.

18. MP-AID LOAD and MP-AID RELOAD Datasets:

    If an MP-AID LOAD or RELOAD command is to be issued in the course of a run, then this additional FILEDEF command is needed to define the input dataset. The specified ddname must be MPAIDR.

19. Concatenated MP-AID:

    An additional FILEDEF command is required for each secondary read-only MP-AID that is to be accessed. The ddname specified must be the same as that given on the MP-AID CONCATENATION command used to establish the required concatenations. Full details of the commands required to implement the concatenated MP-AID are given in the *ASG-Manager Products Systems Adiministrator's Guide*.

20. AUDIT ONTO Output Dataset:

    If an AUDIT ONTO command is to be issued in the course of a run, then an additional FILEDEF command is needed to define the output dataset.

    There is no default ddname assumed. The required ddname is provided as part of the command specification.

21. ADW/IEW Import/Export Datasets:

    A number of additional FILEDEF commands must be provided when executing the ADW/IEW Import/Export facility. The ddnames shown are the defaults and may be tailored by the user.

    You should refer to the *ASG-Manager Products Tools Support: Integration with ADW/IEW* for full details of user tailoring.

22. Generic Import Input Dataset:

    An additional FILEDEF command is required to define a sequential or partitioned dataset used as the input to the extract stage of Generic Import.

    No default ddname is assumed. The required ddname is specified using the EXTERNAL-FILE clause of the EXTRACT command.

23. Procedures Language Trace Dataset:

    An additional FILEDEF command is required if Procedures Language trace output is to be written to a dataset on disk. The default DDNAME for this dataset is MPTRACE, but it may be varied by use of the SET TRACE command.

24. REPLAY Datasets:

    Additional FILEDEF commands are required if the REPLAY facility is to be utilized. The FILEDEF for ddname MPSPRINT is only needed when the PRINT parameter is used.

11 Introduction to Installing MethodManager

ASG-MethodManager (herein called MethodManager) is the tailorable Repository Driven Application Development Environment for enabling AD/Cycle. It draws upon all existing Manager Products software. In general, MethodManager has been designed specifically for interactive use; however, ToolSet SERVICES (apart from rules 100, 110, and 120) and LifeCycle SERVICES can be enabled in batch as well as interactively.

If you are a new user (that is, you do not already have Manager Products installed) or you are an existing Manager Products user who wishes to set up a completely independent and unrelated MethodManager installation, then you should refer to Chapter 12, "Installing An Initial MethodManager Environment," on page 113 and Chapter 13, "Installing an Operational MethodManager Environment," on page 123.

Chapter 12, "Installing An Initial MethodManager Environment," on page 113 provides the equivalent details of setting up an initial MethodManager installation to that already given in Chapter 2, "Installing an Initial Manager Products Environment," on page 19 for setting up an initial Manager Products environment.

Chapter 13, "Installing an Operational MethodManager Environment," on page 123 provides a summary of the tasks required to set up a MethodManager operational installation. In general, the tasks are the same as those described to set up a Manager Products operational installation; any differences are highlighted in Chapter 13.

If you are an existing Manager Products user who wishes to transfer the contents of an existing Manager Products MP-AID and/or repositories to a separate MethodManager installation, then you should also refer to Chapter 14, "Setting up an MP-AID and/or Repositories from an Existing Manager Products Installation," on page 127 for a summary of tasks required to achieve this.

ASG recommends that you become familiar with this manual in general also applicable to MethodManager. You should also refer to the *ASG-MethodManager Administration*.

12 Installing An Initial MethodManager Environment

Introduction

This chapter describes how an initial MethodManager installation can be set up (that is, without considering the effect of functional units or installation macros) in a CMS environment. This initial installation enables new users to experiment and gain a degree of familiarity with MethodManager before an operational MethodManager installation is set up. It consists of these components:

- An executable version of your MethodManager software (supplied in dataset MP.CMS)
- The Manager Products Administration and Information Dataset (MP-AID)
- The ASG-supplied InfoBank (supplied in dataset MP.INFO.UNLOAD and loaded onto the MP-AID)
- The ASG-supplied MethodManager software (supplied in dataset MP.MMR.UNLOAD and loaded onto the MP-AID)
- The ASG-supplied COMMAND members (supplied in dataset MP.COM.UNLOAD and loaded onto the MP-AID)
- The MethodManager SAMPLE Repository (supplied in dataset MP.MMR.SAMPLE).

The setting up of an initial installation is described in six steps:

Step 1. Install an executable version of MethodManager.

Step 2. Set up an MP-AID.

Step 3. Create and restore the MethodManager SAMPLE repository.

Step 4. Run MethodManager under CMS.

Step 5. Set up mainframe support for your PWS Graphical Workbench.

Step 6. Install your PWS Graphical Workbench.

After you have followed these steps, ASG recommends using the MethodManager SAMPLE repository interactively to develop your knowledge of MethodManager and its usage.

The following assumptions have been made in this chapter:

- The MethodManager software has been installed on the 191 disk of the CMS logon ID identified here as MPR1.
- The MP-AID is to be created on the 191 disk of the CMS logon ID identified here as MPR2 and is formatted with a blocksize of 4,096 bytes.
- The SAMPLE repository is to be created on the 191 disk of the CMS logon ID identified here as MPR3 and is formatted with a blocksize of 4,096 bytes.
- The MP-AID and the SAMPLE repository, when created, are NOT configured for access for more than one user. Should you wish to configure either the MP-AID or the SAMPLE repository for concurrent usage, you should follow the instructions given in Chapter 8, "Step 6 - Satisfy Concurrent Usage Requirements," on page 89 of this publication, which describes in full the steps necessary to achieve such usage. If concurrent usage is not implemented and the MP-AID or SAMPLE repository is inadvertently used concurrently, it is likely that both the MP-AID and the SAMPLE repository will be corrupted.
- Installation option 3 has been chosen when installing an executable version of MethodManager.

Step 1 - Install an Executable Version of MethodManager

The release tape containing the MP.CMS dataset should be mounted on a tape unit which has been attached at virtual address 181 to the appropriate CMS logon (in this case logon ID MPR1). ASG recommends that the CMS disk is formatted with a blocksize of 2,048 bytes. Dataset MP.CMS contains the MethodManager CMS software and requires 7,000 2K blocks of disk storage. However, this is the basic disk storage requirement and a further 1,500 2K blocks of disk storage will be required to install an executable version of the MethodManager software.

The Manager Products software is copied to the CMS mini-disk using these CMS commands:

```
TAPE REW
TAPE FSF 40
TAPPDS * TEXT A1 (UPDATE COL1
```

The CMS installation EXECs, supplied by ASG to simplify installation, should now be made available by entering these CMS commands:

```
RENAME MPX0 TEXT A1 MPX0 EXEC A1
MPX0
```

ASG then recommends that you copy the example EXECs MMXI1, MMXI2, MMXI3, MMXI4, and MMXI5 to the CMS system disk or to an extension of the CMS system disk where they can be amended as necessary and made available to all users.

There are 3 installation options for installing MethodManager in a CMS environment. This document assumes that option 3 is to be selected. (Information concerning all options is given in Chapter 4, "Step 2 - Select and Generate an Executable Version of Manager Products," on page 31). Option 3 involves the creation of executable modules in GENMOD Mode, that is, using the CMS GENMOD command. To achieve this you should issue these commands:

```
MPX3
MPX7
```

After successful completion of the above commands, the datasets MP.INFO.UNLOAD, MP.COM.UNLOAD and MP.MMR.UNLOAD must be loaded onto the MP-AID (see"Step 2 - Set Up the MP-AID" on page 115) before the MethodManager software is available for execution.

Step 2 - Set Up the MP-AID

Step 2A - Create an MP-AID and Load Dataset MP.INFO.UNLOAD

Before MethodManager can be used, an MP-AID must be created and the contents of datasets MP.INFO.UNLOAD, MP.COM.UNLOAD, and MP.MMR.UNLOAD loaded.

The dataset named MP.INFO.UNLOAD on the release tape contains an unloaded MP-AID which holds the ASG-supplied InfoBank.

To start, enter these commands:

```
DEFINE STORAGE 3M
IPL CMS
```

You should then run a CMS EXEC containing the CMS commands given in Figure 10. Definitions of the variables used can be found in "Step 4A - Create a BDAM-Organized MP-AID and Load Dataset MP.INFO.UNLOAD" on page 76.

Figure 10. Creating and Loading a BDAM MP-AID with Dataset MP.INFO.UNLOAD

```
/*   MMXI1 - Create and Load a BDAM MP-AID with   */
/*   Data Set MP.INFO.UNLOAD                      */
'EXECOS'
'GLOBAL LOADLIB MPRLIB'
'CP LINK MPR1 191 200 R'
'CP LINK MPR2 191 201 MW'
'ACCESS 200 F'
'ACCESS 201 G'
'SET LDRTBLS 10'
'FILEDEF MPIN TERMINAL'
'FILEDEF MPOUT TERMINAL'
'FILEDEF MPRDIAG DISK MPRDIAG LISTING A1'
'FILEDEF MPAID DISK MMR MPAID G6 (XTENT 6800'
'FILEDEF MPAIDR TAP1 SL 5 (BLOCK 9442'
'LOADMOD MPR00'
'START * LINE'
'GLOBAL LOADLIB'
'RELEASE F (DET'
'RELEASE G (DET'
```

Until the MP-AID has been created, you cannot run MethodManager in full screen mode. To create and load the MP-AID, enter the following Manager Products commands. These commands must be entered in line mode.

```
MP-AID CREATE ADMINISTRATOR SYSAD PASSWORD SAD
PHYSICAL-BLOCKSIZE 8192 LOGICAL-BLOCKSIZE 1024;
LOGON SYSAD PASSWORD SAD;
MP-AID BUFFERS 30;
MP-AID LOAD INFOBANK;
MP-AID STATUS;
LOGOFF;
```

The allocation of an increased size MP-AID buffer pool will enhance performance. A minimum allocation of 30 buffers is recommended.

Note:

The number of blocks allocated to the MP-AID is based on the number of blocks required to install the ASG-supplied InfoSystem, COMMAND members, Corporate Executive Routines and UDS tables, together with about 10 percent available for user storage, if required.

Step 2B - Load Dataset MP.COM.UNLOAD Onto the MP-AID

The dataset MP.COM.UNLOAD on the release tape contains some essential software used by MethodManager.

The installation of this dataset is mandatory.

To load MP.COM.UNLOAD you should first enter these commands:

```
DEFINE STORAGE 3M
IPL CMS
```

You should then run a CMS EXEC containing the CMS commands given in Figure 11 on page 117. Definitions of the variables used can be found in "Step 4B - Load the MP-AID With Dataset MP.COM.UNLOAD" on page 78.

Figure 11. Loading a BDAM MP-AID with Dataset MP.COM.UNLOAD

```
/*   MMXI2 - Load the MP-AID with Data Set MP.COM.UNLOAD   */
/*                                                         */
'EXECOS'
'GLOBAL LOADLIB MPRLIB'
'CP LINK MPR1 191 200 R'
'CP LINK MPR2 191 201 MW'
'ACCESS 200 F'
'ACCESS 201 G'
'SET LDRTBLS 10'
'FILEDEF MPIN TERMINAL'
'FILEDEF MPOUT TERMINAL'
'FILEDEF MPRDIAG DISK MPRDIAG LISTING A1'
'FILEDEF MPAID DISK MMR MPAID G6 (XTENT 6800'
'FILEDEF MPAIDR TAP1 SL 4 (BLOCK 9442'
'LOADMOD MPR00'
'START *'
'GLOBAL LOADLIB'
'RELEASE F (DET''RELEASE G (DET'
```

When MethodManager starts executing you will be presented with a Logon Panel. You should enter SYSAD as the Logon Identifier and SAD as the password. Once you have successfully logged on, enter these Manager Products commands:

```
MP-AID BUFFERS 30;
MP-AID LOAD ALL ;
MP-AID STATUS ;
LOGOFF ;
```

Step 2C - Load the MP-AID With Dataset MP.MMR.UNLOAD

The dataset MP.MMR.UNLOAD on the release tape contains the default ToolSet SERVICES, LifeCycle SERVICES, and Strategic Information Planning software.

The installation of this dataset is mandatory.

To load MP.MMR.UNLOAD you should first enter these commands:

```
DEFINE STORAGE 3M
IPL CMS
```

You should then run a CMS EXEC containing the CMS commands given in Figure 12 on page 118.

Figure 12. Loading a BDAM MP-AID with Dataset MP.MMR.UNLOAD

```
/*   MMXI3 - Load the MP-AID with Data Set MP.MMR.UNLOAD   */
/*                                                         */
'EXECOS'
'GLOBAL LOADLIB MPRLIB'
'CP LINK MPR1 191 200 R'
'CP LINK MPR2 191 201 MW'
'ACCESS 200 F'
'ACCESS 201 G'
'SET LDRTBLS 10'
'FILEDEF MPIN TERMINAL'
'FILEDEF MPOUT TERMINAL'
'FILEDEF MPRDIAG DISK MPRDIAG LISTING A1'
'FILEDEF MPAID DISK MMR MPAID G6 (XTENT 6800'
'FILEDEF MPAIDR TAP1 SL 22 (BLOCK 9442'
'LOADMOD MPR00'
'START *'
'GLOBAL LOADLIB'
'RELEASE F (DET'
'RELEASE G (DET'
```

When MethodManager starts executing you will be presented with a Logon Panel. You should enter SYSAD as the Logon Identifier and SAD as the password. Once you have successfully logged on, enter these Manager Products commands:

```
MP-AID BUFFERS 30;
MP-AID LOAD ALL REPLACE ;
MP-AID STATUS ;
LOGOFF ;
```

Step 3 - Create and Restore a Sample Repository

Your MethodManager release tape contains a dataset, MP.MMR.SAMPLE, which holds the SAMPLE repository. This repository can be retrieved by running MethodManager in order to create an empty repository and to restore the SAMPLE repository.

You should firstly enter these commands:

```
DEFINE STORAGE 3M
IPL CMS
```

You should then run a CMS EXEC containing the CMS commands given in Figure 13 on page 119 to create a BDAM repository. Definitions of the variables used can be found in "Set Up Manager Products Administration Dictionary" on page 84.

Figure 13. Creating and Restoring a BDAM-Organized SAMPLE Repository.

```
/*   MMXI4 - Create and restore a BDAM SAMPLE Repository   */
/*                                                         */
'EXECOS'
'GLOBAL LOADLIB MPRLIB'
'CP LINK MPR1 191 200 R'
'CP LINK MPR2 191 201 MW'
'CP LINK MPR3 191 202 MW'
'ACCESS 200 F'
'ACCESS 201 G'
'ACCESS 202 H'
'SET LDRTBLS 10'
'FILEDEF MPIN TERMINAL'
'FILEDEF MPOUT TERMINAL'
'FILEDEF MPRDIAG DISK MPRDIAG LISTING A1'
'FILEDEF MPAID DISK MMR MPAID G6 (XTENT 6800'
'FILEDEF SAMPLE DISK SAMPLE INDEX H6 (XTENT 15'
'FILEDEF SAMPLED DISK SAMPLE DATAENT H6 (XTENT 300'
'FILEDEF SAMPLES DISK SAMPLE SOURCE H6 (XTENT 500'
'FILEDEF SAMPLEE DISK SAMPLE RECOVER H6 (XTENT 50'
'FILEDEF SAMPLEJ DISK SAMPLE LOG H6 (XTENT 50'
'FILEDEF SAMPLER TAP1 SL 21 (BLOCK 9442'
'LOADMOD MPR00'
'START *'
'GLOBAL LOADLIB'
'RELEASE F (DET'
'RELEASE G (DET'
'RELEASE H (DET'
```

When MethodManager starts executing you will be presented with a Logon Panel into which you should enter SYSAD for the Logon Identifier and SAD for the password. Once you have successfully logged on you should enter these Manager Products commands:

```
CREATE SAMPLE MASTER CON
ILB 2046 SLB 314 DLB 372 IPB 8192 SPB 4096 DPB 8192
RPB 4096 LPB 4096 WITH 7 STATUSES AND LOG;
DICTIONARY SAMPLE;
AUTHORITY CON;
RESTORE ALL;
LOGOFF;
```

Step 4 - Run MethodManager Under CMS

You should firstly enter these commands:

```
DEFINE STORAGE 5M
IPL CMS
```

You should then run a CMS EXEC containing the CMS commands given in Figure 14. Definitions of the variables used can be found in Chapter 10, "Step 8 - Run Manager Products with BDAM MP-AID and BDAM Dictionaries," on page 101.

Figure 14. Running MethodManager (BDAM MP-AID and Repository)

```
/*   MMXI5 - Running METHODMANAGER (BDAM MP-AID and Repository)   */
/*                                                                */
'EXECOS'
'GLOBAL LOADLIB MPRLIB'
'CP LINK MPR1 191 200 R'
'CP LINK MPR2 191 201 MW'
'CP LINK MPR3 191 202 MW'
'ACCESS 200 F'
'ACCESS 201 G'
'ACCESS 202 H'
'SET LDRTBLS 10'
'FILEDEF MPIN TERMINAL'
'FILEDEF MPOUT TERMINAL'
'FILEDEF MPRDIAG DISK MPRDIAG LISTING A1'
'FILEDEF MPRT DISK MPRT LISTING A1'
'FILEDEF MPAID DISK MMR MPAID G6 (XTENT 6800'
'FILEDEF SAMPLE DISK SAMPLE INDEX H6 (XTENT 15'
'FILEDEF SAMPLED DISK SAMPLE DATAENT H6 (XTENT 300'
'FILEDEF SAMPLES DISK SAMPLE SOURCE H6 (XTENT 500'
'FILEDEF SAMPLEE DISK SAMPLE RECOVER H6 (XTENT 50'
'FILEDEF SAMPLEJ DISK SAMPLE LOG H6 (XTENT 50'
'LOADMOD MPR00'
'START *'
'GLOBAL LOADLIB'
'RELEASE F (DET'
'RELEASE G (DET'
'RELEASE H (DET'
```

Step 5 - Set Up Mainframe Support for Your PWS Graphical Workbench

To set up the mainframe Executive Routines that support your PWS Graphical Workbench, in the correct form, you must enter this command once only:

```
MVW-GEN INSTALL;
```

Step 6 - Install Your PWS Graphical Workbench

Please refer to the *ASG-ManagerView User's Guide* for details of this installation step.

13 Installing an Operational MethodManager Environment

General Requirements

The following steps are required to perform an operational MethodManager installation in a CMS environment. You should note that the step numbers are consistent with those used in this publication.

Steps Required to Install MethodManager in a CMS Environment

Step Number	Step Name	References	Notes
1	Copy datasets to disk	Chapter 3, "Step 1 - Copy Datasets to Disk," on page 27	-
2	Select and generate an executable version of Manager Products	Chapter 4, "Step 2 - Select and Generate an Executable Version of Manager Products," on page 31	Additionally you must execute CMS EXEC MPX7. Note that 1,500 2K blocks of disk storage are required instead of 1,000 as specified in Chapter 4, "Step 2 - Select and Generate an Executable Version of Manager Products," on page 31.
3	Tailor Manager Products using the installation macros and modules in dataset MP.CMSRCE	Chapter 5, "Step 3 - Tailoring Installation Macros and Modules Supplied in MP.CMSRCE," on page 37	-
4	CREATE an MP-AID		-
A	Create an MP-AID and load dataset MP.INFO.UNLOAD	Chapter 6, "Step 4 - Create and Load the MP-AID," on page 75	The presence of InfoBank is mandatory for the successful functioning of and ongoing integrity of MethodManager.
B	Load the MP-AID with dataset MP.COM.UNLOAD	Chapter 6, "Step 4 - Create and Load the MP-AID," on page 75	-
C	Load the MP-AID with dataset MP.MMR.UNLOAD	Chapter 6, "Step 4 - Create and Load the MP-AID," on page 75	-
5	Set up repositories	Chapter 7, "Step 5 - Set Up Dictionaries," on page 81	-

Steps Required to Install MethodManager in a CMS Environment

Step Number	Step Name	References	Notes
A	Set up Administration Repository	"MethodManager Administration Repository" on page 125	-
6	Satisfy concurrent usage requirements	Chapter 8, "Step 6 - Satisfy Concurrent Usage Requirements," on page 89	-
7	Link external software to Manager Products software	Chapter 9, "Step 7 - Linking External Software To Manager Products Software," on page 95	-
8	Run MethodManager with BDAM MP-AID and BDAM repositories	Chapter 10, "Step 8 - Run Manager Products with BDAM MP-AID and BDAM Dictionaries," on page 101	See note 1, note 2, and note 3
9	Set up mainframe support for your PWS Graphical Workbench		Enter this command once only: `MVW-GEN INSTALL;`
10	Install your PWS Graphical Workbench	*ASG-ManagerView User's Guide*	-

Note:

1. It is necessary to specify a ddname of MPRT for the alternative output dataset.

2. You should add a minimum of 1,000K and a maximum of 2,000K to the program code virtual storage requirements specified in Appendix E,"Virtual Storage Requirements for Running Manager Products," on page 157, depending upon the facilities to be used.

3. If the MethodManager HARDCOPY command is used, then it is necessary to provide a CMS batch machine that is capable of executing MethodManager batch jobs submitted by MethodManager interactive users. The requirements for the disconnected batch machine are:

   - It should have minimum a virtual machine size of 1MB.
   - The 195 A-disk must be large enough to hold the one or two LISTING files which are generated. ASG recommends that a minimum of 5 cylinders (on a 3,380 disk) should be allocated for this disk. This should be sufficient for most situations.

Upon completion of each batch job the generated LISTING files are spooled to the virtual reader of the submitting user. You do not need to define a disconnected batch machine for the exclusive use of the MethodManager users, but if a high volume of MethodManager batch jobs is anticipated, then you may prefer to dedicate a batch machine for MethodManager use only.

You must ensure that before using the HARDCOPY command, user-definable parameters of the ASG-supplied skeleton CMS batch job are tailored to your particular installation and the updated skeleton filed as a USER-MEMBER on the MP-AID. Option 9.7.1.1 on the ToolSet SERVICES main menu is used to invoke the customization procedure.

MethodManager Administration Repository

To set up an Administration Repository for use with MethodManager you should perform these four steps:

1. Create an empty repository in the same way as any other Manager Products repository.

2. Restore the ASG-supplied repository MP.MMR.ADMIN from the installation tape into the empty repository by using the Controller's RESTORE ALL command. The job control requirements for achieving this are those job control statements that are specified as being compulsory for all users in Table 17 on page 102 , plus this statement:

   ```
   FILEDEF dictR TAP1  SL n (BLOCK 9442
   ```

   where:

   dict is the name of the empty repository that will become the MethodManager Administration Repository.

   n is the position of dataset MP.MMR.ADMIN on the release tape. The position of the dataset is given in the list of datasets provided with this tape.

   The dataset MP.MMR.ADMIN is a SAVE ALL of the MethodManager software. It contains six frozen statuses named MDRIM2200, ADMIN2200, MDRIM2210, ADMIN2210, MDRIM2300, and ADMIN2300.

   The status MDRIM contains the repository definitions for the MethodManager Information Models DU016 and DU777.

   The statuses ADMIN contains the ToolSet SERVICES (TSS) and LifeCycle SERVICES (LCS) tailorable software.

3. You may also wish to restore the ASG-supplied Executive Routines (supplied in dataset MP.CORP). See "Restore ASG-supplied InfoDictionary" on page 86 for further details.

4. You may also wish to restore the ASG-supplied SAVEd source of Infobank (supplied in dataset MP.INFO). See "Restore ASG-supplied InfoDictionary" on page 86 for further details.

See Appendix C,"Attributes of Datasets Which May Be Supplied on Your Release Tape," on page 149 for attributes of all ASG-supplied datasets.

See Appendix D,"Attributes of ASG-supplied Dictionaries," on page 153 for attributes of ASG-supplied repositories.

Installing the MethodManager Strategic Information Planning Life Cycle

In order to utilize the MethodManager Strategic Information Planning (SIP) Life Cycle you need to restore and encode the members which comprise the Life Cycle into the root status of your production repository.

The SIP Life Cycle is supplied in status LIFE-CYCLE of the dataset MP.MMR.SAMPLE and can be restored using the command:

```
RESTORE SOURCE FROM LIFE-CYCLE;
```

After encoding the restored members the SIP Life Cycle can be activated. For further details refer to the *ASG-MethodManager Strategic Information Planning*.

14 Setting up an MP-AID and/or Repositories from an Existing Manager Products Installation

If you have already set up a Manager Products installation, then you may wish to set up a separate MethodManager installation which uses repositories and an MP-AID from your existing Manager Products installation. You may, with the same MP-AID, have some repositories configured for use with Manager Products, MethodManager ToolSet SERVICES, or MethodManager LifeCycle SERVICES and some repositories configured for use under Strategic Information Planning. The RIM (that is, UDS table) needs to be changed to DU016 for each repository which is to be converted for use with MethodManager Strategic Information Planning.

MethodManager ToolSet SERVICES will operate with any RIM providing that ToolSet SERVICES have been set up as described in *ASG-MethodManager Administration*.

MethodManager LifeCycle SERVICES will operate with any RIM providing that you incorporate the member type groups supplied in the MethodManager Administration repository into your RIM structure.

These steps need to be performed:

- Install the new version of the Manager Products software containing MethodManager supplied in data set MP.CMS.
- Load an UNLOADed copy of your MP-AID into a newly created MP-AID.
- Load data set MP.MMR.UNLOAD onto the MP-AID from the Release tape.
- Load a SAVEd copy of each of your repositories into the newly created repository.
- For each Manager Products repository to be used with MethodManager Strategic Information Planning, convert to RIM DU016.
- Set up a MethodManager Administration Repository, as described in "MethodManager Administration Repository" on page 125.

Appendix A
The Master Operator

A dictionary Controller may choose to delegate some functions to a member of the Operations department. The functions that can be delegated are those that are concerned with the physical back-up and recovery of the dictionary; they do not involve any reorganization of or access to the contents of the dictionary. The person to whom these functions are delegated is known in ASG terminology as the Master Operator.

The Master Operator is allocated a special password by the Controller. This password can be quoted in AUTHORITY commands, in RELOAD and ROLL-FORWARD commands, and in free-standing versions of LOG ANALYSIS and LOG STATUS command; all include a dictionary name and password. Your Manager Products check that the password quoted is the Controller's password or the Master Operator's password for the dictionary named in the command.

These are the commands available to the Master Operator:

- AUTHORITY
- DICTIONARY
- DISABLE
- ENABLE
- JOURNAL
- LOGON
- LOGOFF
- LOG ANALYSIS
- LOG ARCHIVE
- LOG BACKUP-DETAILS
- LOG STATUS
- LOG SWITCH
- RELOAD
- ROLL-FORWARD
- UNLOAD

Except when free-standing commnads (those that include a dictionary name and password in the command itself) are being used, the Master Operator first opens the dictionary by a DICTIONARY command, then issues an AUTHORITY command, quoting the specially allocated password. Once this is accepted, any of the other commands listed above may be issued.

The Master Operator should receive specific job instructions from the Controller for the use of the available commands, in accordance with the security and operational standards in force at the particular installation.

Appendix B

Environments Supported by Manager Products

Particular Manager Products interface with certain IBM program products, and/or Siemens program products, and/or other vendor program products that run within one or more of these operating system environments:

- IBM OS
- IBM DOS
- IBM VM/CMS
- SIEMENS BS2000
- MS-DOS
- PC-DOS
- OS/2

Host Support

Manager Products host-based software may run on IBM System/370, IBM System/390, or plug-compatible computers that support the operating systems described in this Appendix.

For Mainframe Environment (MFE)-based Manager Products, ASG provides Maintenance and Updating service in IBM OS, DOS, VM/CMS, and Siemens BS2000 environments. ASG provides Maintenance and Updating Service for MFE-based elements of MethodManager in IBM OS and VM/CMS environments (DOS and BS2000 support is yet to be announced for MethodManager).

ASG's Maintenance and Updating Service for a particular MFE OS, DOS VM/CMS, or BS2000 environment (compatible Version/Release Level) will continue for a period equal to or greater than the IBM or Siemens support for that environment.

The tables given later in this Appendix show the Manager Products that support a particular environment/product: the Compatible Release Level(s) at which the products operate with the relevant Manager Products are also shown:

ControlManager and DictionaryManager are co-requisites of each other within the ASG-Manager Family of Program Products (excluding MethodManager). Both are Environmental Prerequisites (EPR) is as much as they must be at the latest ASG Version and Release Level for each and every other Manager Products to execute correctly. This EPR rule applies to Manager Products in both Mainframe Environments (MFE) and Programmable Workstation Environments (PWSE). PWSE is also known as Intelligent Workstation Environments (IWSE).

ControlManager and DictionaryManager complement each other in providing a gateway environment to OSI (Open Systems Interconnection) across information engineering techniques and dictionaries/directories/repositories from ASG and other vendors.

Thus, ControlManager and DictionaryManager enable Manager Products users to position themselves to take full advantage of the Manager Products Family in providing a Computer Aided Software Engineering (CASE) environment.

You should consult the appropriate ASG technical publications for further details of the nature and extent of Manager Products functional support provided.

It is possible that the release support and/or the functionality provided for a given environment/Program Product may be dependent on or enhanced by the presence of one or more selectable units and/or another Manager Products, for example:

- Use of the ControlManager Extended Interactive Facility (CMR-FE01) is dependent on the presence of the CICS Interface (CMR-TP2) or the OS/TSO Interface (CMR-TP7) or the VM/CMS Interface (CMR/TP8), or the Siemens Time Sharing Interface (CMR-TP11).
- The functionality offered by User Defined Commands (CMR-UDOS) is considerably enhanced if the Extended Interactive Facility (CMR-FE01) is also installed.

Host Terminals Supported

IBM 31*xx* and 327*x* and compatible terminals, and Siemens 8160 and compatible terminals, are supported. If IBM or Siemens introduces any other non-programmable terminals (NPTs) subsequent to the date of publication of this Appendix, you should contact the ASG Service Desk to verify support for such NPT.

PWS Support

Manager Products PWS-based software may be run on IBM plug-compatible programmable workstations that support the operating environments described in this Appendix. For Programmable Workstation Environment (PWSE)-based Manager Products, ASG provides Maintenance Subscription and Updating Service in Windows 95 and Windows NT environments. The tables given later in this Appendix show the Manager Products that support a particular environment product; The Compatible Release Level(s) at which the products operate with the relevant Manager Products are also shown.

The functionality of ASG-ManagerView (herein called ManagerView) is available:

- As part of the Programmable Workstation (PWS) Product within MethodManager.
- As part of the Intelligent Workstation Environment (IWSE) Package and Host Connectivity within the ASG-Manager Family of Program Products.

For a definition of the complete PWS Package, please refer to the current (or any updates subsequently published by ASG) Terms and Conditions of the MethodManager Order.

For a definition of the IWSE Package see "The IWSE Order for an ASG-IWSE Package - Attachment 1" on page 144.

For hardware and software prerequisites of the PWS Package please refer to the current (or any updates subsequently published by ASG) "The MMR Order for a PWS Package - Attachment 1" on page 142.

For hardware and software prerequisites of the IWSE Package please refer to the current (or any updates subsequently published by ASG) "The IWSE Order for an ASG-IWSE Package - Attachment 2" on page 146.

Host Environments

Table 18. Environments in which Manager Products may be Used (for MethodManager see separate table on page 135)

Host Environment		Manager Products				
Name (Note 5)	Compatible Version/Release Level	CMR Version 02 Release 4.0	DMR Version 01 Release 10.0	DSR Version 01 Release 4.3	DYR Version 02 Release 4.0	MVW CMR (Note 8)
Under OS						
MVS/SP batch	All	Yes	Yes	Yes	Yes	No
MVS/ESA batch	All	Yes	Yes	Yes	Yes	No
MVS/XA batch	All	Yes	Yes	Yes	Yes	No
OS/390 batch	All	Yes	Yes	Yes	Yes	No
CICS/VS	1.6 to 4.1	Yes	Yes (Note 6)	No	Yes	Yes
CMS (Note 9)	All	Yes	Yes	Yes	Yes	Yes
IMS/DC	1.1.4 to 1.3, (2.0), (4.1)	Yes	Yes (Note 6)	No	Yes	No
ISPF (TSO)	All	Yes	Yes	Yes	Yes	Note 7 and Note 14
ROSCOE (Note 11)	5.4 to 5.6	Yes	Yes	Yes	Yes	No
TSO	All	Yes	Yes	Yes	Yes	Yes
TSO/E	All	Yes	Yes	Yes	Yes	Yes
Under DOS						
VSE batch	All	NYA	NYA	NYA	NYA	No
VSE/SP batch	All	NYA	NYA	NYA	NYA	No
SSX/VSE batch	(All)	NYA	NYA	NYA	NYA	No
CICS/VS	1.6, 1.7	NYA	NYA	NYA	NYA	Yes
CMS (Note 9)	All	NYA	NYA	NYA	NYA	Yes
ICCF	(All)	NYA	NYA	NYA	NYA	No
Under VM (native)						
CMS (Note 9)	(All)	NYA	NYA	NYA	NYA	Yes

Table 18. Environments in which Manager Products may be Used (for MethodManager see separate table on page 135)

Host Environment		Manager Products				
Under Siemens						
BS2000 batch	To be supplied	NYA	NYA	NYA	NYA	No
Siemens Timesharing Monitor (Note 10)	See BS2000	NYA	NYA	NYA	NYA	NYA (Note 13)

Note:

NYA=Not Yet Available
For notes see page 137

Table 19. Environments in which MethodManager may be Used

Host Environment		MethodManager Version 02 Release 4.0	
Name (Note 5)	Compatible Version/Release Level	PWS Resident Component	Host Resident Component
Under OS			
MVS/SP batch	All	N/A	Yes
MVS/ESA batch	All	N/A	Yes
MVS/XA batch	All	N/A	Yes
OS/390 batch	All	N/A	Yes
CICS/VS	1.6 to 4.1	NYA	NYA
CMS (Note 9)	All	Yes	Yes
IMS/DC	l.l.4 to l.3, (2.0), (4.1)	No	No
ISPF (TSO)	All	Note 7	Note 7
ROSCOE	5.4 to 5.6	No	No (Note 12)
TSO	All	Yes	Yes
TSO/E	All	Yes	Yes
Under DOS			
VSE batch	All	N/A	NYA
VSE/SP batch	All	N/A	NYA
SSX/VSE batch	(All)	N/A	NYA
CICS/VS	1.6, 1.7	NYA	NYA

Table 19. Environments in which MethodManager may be Used

Host Environment		MethodManager Version 02 Release 4.0	
CMS (Note 9)	All	NYA	NYA
ICCF	(All)	No	No
Under VM (native)			
CMS (Note 9)	(All)	NYA	NYA
Under Siemens			
BS2000 batch	To be supplied	No	NYA
Siemens Timesharing Monitor (Note 10)	See BS2000	NYA	NYA

Note:

NYA=Not Yet Available
N/A=Not Applicable
For notes see page 137

PWS/IWSE Environments: Third Party Environmental Operating Requirements

Table 20. MethodManager Version 02 Release 4.0 - PWS/Host Compatibility

PWS Environment		Compatible Host Environment				
Name		OS/TSO and OS/TSO/E	OS/CICS	VM/CMS	DOS/CICS	Siemens Timesharing Monitor
	Compatible Version/Release Level	Any	1.6 to 4.1	Any	1.6, 1.7	To be supplied
PC-DOS or MS-DOS plus Microsoft Windows	Version 3.1 and subsequent Versions 3.1 and 3.11	Yes	NYA	Yes	NYA	NYA
OS/2 (utilizing OS/2 Windows support)	Release 2.1	Yes	NYA	Yes	NYA	NYA

Table 21. ManagerView Version 01 Release 5.0 - IWSE/Host Compatibility

IWSE Environment		Compatible Host Environment				
Name		OS/TSO and OS/TSO/E	OS/CICS	VM/CMS	DOS/CICS	Siemens Timesharing Monitor
	Compatible Version/Release Level	Any	1.6 to 4.1	Any	1.6, 1.7	To be supplied
Microsoft Windows 95 or later Microsoft NT 3.51 or later		Yes	Yes	Yes	Yes	NYA

Note:

NYA = Not Yet Available
For notes see page 137

Notes for the Tables on Pages 134 to 137

1. Product Releases of other vendor Program Products that are shown in parentheses () have yet to be tested fully for compatibility with Manager Products; however, ASG expects at a future time to confirm support.

2. Where All is specified, all Releases of the relevant other vendor Program Products that are available and supported by the vendor at the Date of Publication of this Appendix are compatible with the stated Manager Products Releases. ASG will not necessarily automatically support new Releases issued by the other vendor between the Date of Publication of this Appendix and its next update.

3. Between the date of this publication and its next update ASG may:

   - Introduce support for environments that were not available or were not supported by ASG at the date of publication
   - Withdraw support for environments that were available or were supported by ASG on the date of publication.

   To verify the most up-to-date situation, contact the ASG Service Desk.

4. The Manager Products listed do not necessarily support every feature of every version of other vendor Program Products.

5. In environments that support 31-bit addressing, such as MVS or OS/390, the Manager Products program code will be loaded into both 24- and 31-bit virtual storage. Dynamically allocated storage is obtained mainly from 31-bit virtual storage. The addressing mode of Manager Products is therefore both 24- and 31-bit.

Environment-specific Notes

6. The Siemens UDS Generation (DMR-SL10) and the SESAM Generation (DMR-SL11) selectable units are not available for use under CICS/VS or IMS/DC.

7. This Program Product can be invoked using ISPF services (under TSO); however it does not utilize ISPF's interactive capabilities.

8. This represents ManagerView Version 01 Release 5.0, coupled with ControlManager Version 02 Release 4.0 through the ControlManager Workstation Interface - PC and Mainframe Tailoring (selectable unit CMR-WS01).

9. When executing Manager Products under CMS, only functions of native VM and CMS are utilized; this is true whether or not OS or DOS systems are also running on the machine on which VM is installed. All Releases of CMS available and supported by IBM at the Date of Publication of this Appendix are compatible with the stated Manager Products Releases.

10. The Siemens Timesharing Monitor interface also supports TOMTI Releases 2.0, 2.2B, and 2.2C. (Not yet confirmed at the date of publication.)

11. ROSCOE users who wish to run ManagerView may do so by using the ROSCOE ETSO facility, which effectively provides a TSO interface.

12. ROSCOE ETSO users have the alternative of ordering and using the OS/TSO Interface (CMR-TP7).

13. ManagerView, when used in conjunction with the host Siemens Timesharing Monitor, currently links through ManagerView Import/Export rather than through host connectivity.

14. ManagerView supports the ISPF interface (CMR-FE70) from ControlManager Version 02 Release 2.0 and later.

Key

N/A	Not Applicable
NYA	Not Yet Available
CMR	ControlManager
DMR	DataManager
DSR	DesignManager
DYR	DictionaryManager
MMR	MethodManager
MVW	ManagerView

System and Language Support

Table 22. Systems and Languages with which Manager Products may be Interfaced (for MethodManager see separate table on page 140)

System or Language		Manager Products				
Name	Compatible Version/Release Level	CMR Version 02 Release 4.0	DMR Version 01 Release 10.0	DSR Version 01 Release 4.3	DYR Version 02 Release 4.0	MVW CMR (Note 6)
Program Language Compilers						
Assembler	All	N/A	Yes	N/A	No	N/A
COBOL	All	N/A	Yes	N/A	No	N/A
MARK IV	Up to 7.0	N/A	Yes	N/A	No	N/A
PL/I	All	N/A	N/A	No	No	
Third Party Vendor Platforms						
IEW	From 5.0	Yes	Yes	N/A	Yes	N/A
ADW	From 1.5	Yes	Yes	N/A	Yes	N/A
Database Management Systems						
ADABAS	Up to 4.1	N/A	Yes	No	Yes	N/A
DB2	V01 Rel 3.0 V02 Rel1.0,2.0,3.0 V03 V4.1 V5.1	N/A	Yes	Yes	Yes	N/A
DOS DL/I	1.4 to 1.7	N/A	Yes	Yes	Yes	N/A
IDMS/R and IDD	10.0 3.0, 10.0	N/A	Yes	No	Yes	N/A
IMS/VS(DL/I)	1.1.4 to 1.3, (2.0 to 5.1)	N/A	Yes	Yes	Yes	N/A
SESAM	To be supplied	N/A	NYA (Note 5)	No	NYA	N/A
Siemens UDS	To be supplied	N/A	NYA (Note 5)	No	NYA	N/A
SQL/DS	V02 Rel 1.0 and Rel 2.0	N/A	Yes	Yes	Yes	N/A
System 2000/80	2.9.0	N/A	N/A	Yes	No	N/A
TOTAL/8	All	N/A	Yes	No	Yes	N/A
Program Library Systems						
VSE/SSL	All	N/A	Yes	N/A	Yes	N/A
VSE/LIBR	All	N/A	Yes	N/A	Yes	N/A
The Librarian	All	N/A	Yes	N/A	Yes	N/A
OS/PDS	All	N/A	Yes	N/A	Yes	N/A
PANVALET	All	N/A	Yes	N/A	Yes	N/A

Note:

N/A=Not Applicable
NYA=Not Yet Available
For notes see page 141

Table 23. Systems and Languages with which MethodManager may be Interfaced

System or Language		MethodManager
Name	Compatible Version/Release Level	Version 02 Release 4.0
Program Language Compilers		
Assembler	All	Yes
COBOL	All	Yes
MARKIV	Up to 7.0	Yes
PL/I	All	Yes
Third Party Vendor Platforms		
IEW	From 5.0	Yes
ADW	From 1.5	Yes
Database Management Systems		
ADABAS	Up to 4.1	Note 7
DB2	V01Rel 3.0 V02 Rel 1.0, 2.0, 3.0 V03 V 4.1 V5.1	Yes
DOS DL/I	1.4 to 1.7	Yes
IDMS/R and IDD	10.0, 3.0, 10.0	Note 8
IMS/VS (DL/I)	1.1.1 to 1.3 (2.0 to 5.1)	Yes
SESAM	To be supplied	Note 7
Siemens UDS	To be supplied	Note 7
SQL/DS	V02 Rel 1.0 and Rel 2.0	Yes
System 2000/80	2.9.0	Note 8
TOTAL/8	All	Note 8
Program Library Systems		
VSE/SSL	All	Yes
VSE/LIBR	All	Yes

Table 23. Systems and Languages with which MethodManager may be Interfaced

System or Language		MethodManager
The Librarian	All	Yes
OS/PDS	All	Yes
PANVALET	All	Yes

Note:

For notes see page 141

Notes for the Tables on pages 139 and 140

1. Product Releases of other vendor Program Products that are shown in parentheses () have yet to be tested fully for compatibility with Manager Products; however, ASG expects at a future time to confirm support.

2. Where All is specified, all Releases of the relevant other vendor Program Products that are available and supported by the vendor at the date of publication of this Appendix are compatible with the stated Manager Products Releases. ASG will *not* necessarily automatically support new Releases issued by the other vendor between the date of publication of this Appendix and its next update.

3. Between the date of this publication and its next update ASG may:

   - Introduce support for environments that were not available or were not supported by ASG at the Date of Publication
   - Withdraw support for environments that were available or were supported by ASG on the Date of Publication

   To verify the most up-to-date situation, contact the ASG Service Desk.

4. The Manager Products listed do not necessarily support every feature of every version of other vendor Program Products.

Environment-specific Notes

5. The Siemens UDS Generation (DMR-SL10) and the SESAM Generation (DMR-SL11) selectable units are not available for use under CICS/VS or IMS/DC.

6. This represents ManagerView Version 01 Release 5.0, coupled with ControlManager Version 02 Release 4.0 through the ControlManager Workstation Interface - PC and Mainframe Tailoring (selectable unit CMR-WS01).

7. This interface is ALIGNED; that is, its functionality, though not yet migrated into MethodManager, is expected to be provided in a future MethodManager Version/Release. Meanwhile, the corresponding interfaces through DataManager and/or ControlManager may be attached to MethodManager to satisfy this requirement.

8. This interface is UNADOPTED; that is, ASG does not presently intend to migrate its functionality into MethodManager. However, the corresponding interfaces through DataManager and/or ControlManager may be attached to MethodManager to satisfy this requirement.

Key

N/A	Not Applicable
NYA	Not Yet Available
CMR	ControlManager
DMR	DataManager
DSR	DesignManager
DYR	DictionaryManager
MMR	MethodManager
MVW	ManagerView

The MMR Order for a PWS Package - Attachment 1

3rd Party Environmental Operating Requirements

Use of the PWS/IWSE Package requires that one of the mandatory prerequisite hardware configurations of the following 3rd Party Vendors be met:

Table 24. PWS/IWSE Software Environmental Prerequisite Configurations (see note 9)

Operating Environment:
Microsoft Windows NT 3.51 or later
Microsoft Windows 95 or later

Table 25. PWS/IWSE Hardware Environmental Prerequisite Configurations

Component	Windows
Processor	80486(80386 preferred)
Minimum RAM	16 MB
Display	any supported by Windows
Disks	Hard disk (see Note 8) high-density 3.5
Mouse	any supported by Windows
Host Connectivity	see Table 24

Table 26. PWS/IWSE Host Connectivity

	3270 Emulation Product (see Note 7)	Minimum Version
1.	Attachmate EXTRA! for Windows	Version 4.1.0
2.	Banyan Vines Extended SNA Gateway	Version 1.3
3.	Attachmate IRMA WorkStation for Windows	Version 3.0
4.	IBM Personal Communications/3270 for Windows	Version 4.0.0
5.	IBM Communications Manager/2 (OS/2)	Version 1.11
6.	Novell Netware 3270 LAN Workstation	Version 1.1
7.	Wall Data Rumba for Mainframe	Version 4.0.0

Notes of Explanation

1. ASG cannot and does not represent that its Manager Products will be compatible with any combination of non-ASG products you choose to use them with. While your ASG product supplier may be able to help, you must determine for yourself the compatibility in any particular instance of Manager Products and your hardware/software environment.

2. Your attention is drawn to Table 24, PWS/IWSE Host Connectivity. The products listed may themselves have certain 3rd Party Host prerequisites. Your ability to utilize your 3270 Emulation Product must be verified with your systems programming department, who should be requested to review all the relevant documentation.

3. The tables shown represent ideal Prerequisite Configurations that can be modified by you at your discretion provided you maintain 100 percent compatibility. Your major concern should be to ensure that your installation conforms to the definition of the 3rd Party Software and 3rd Party Hardware that provides the Host Connectivity.

4. Between the date of this publication and its next update ASG may:

   - Introduce support for environments that were not available or ware not supported by ASG at the date of publication
   - Withdraw support for environments that were available or were supported by ASG on the date of publication.

   To verify the most up-to-date situation, contact the ASG Service Desk.

5. ASG acknowledges that certain proprietary products or services or programs mentioned within this publication or within other ASG publications are distributed under Trademarks or Registered Trademarks of the vendors who own and/or distribute the products or services or programs in the country in question.

6. The information contained herein is subject to change without notice.

7. Table 24 lists certain 3270 Emulation products which provide HLLAPI and/or LLAPI programming interfaces. The ASG PWS/IWSE Packages, MethodManager PWS Graphical Workbench (PGW), and ManagerView (MVW), communicate through these programming interfaces. You will require specific hardware in your PWS/IWSE which is specified by the vendor of the 3270 Emulation product you choose to use. From a PGW/MVW viewpoint the hardware you use to connect to the host may not be significant, and it is your decision to use any connection supported by the 3270 Emulation product. Typical connections include co-ax, network using a gateway, synchronous, or asynchronous using modems, etc.

8. A minimum of 5MB of hard disk space is required to install and execute the PWS/IWSE Package; however it is recommended that 10 MB of space be available to provide sufficient expansion for application use of the product.

9. Local Area Networks:

   ASG PWS/IWSE products, PGW and MVW, can be run on a Local Area Network as follows:

   - The PWS/IWSE software can be stored on a network drive, providing a consistent version of the products to all users on the LAN
   - The host connections can be routed through a network gateway
   - The Local (PWS/IWSE) Dictionary can be stored on a network drive so that any user with access to the LAN (standard security arrangements permitting) can access it. (Concurrent access by multiple users will be a supported feature in a subsequent Manager Products release.) Currently it is possible for multiple users to have access to the local (PWS/IWSE) Dictionaries, but such use may result in corruption of the content.

The IWSE Order for an ASG-IWSE Package - Attachment 1

ASG Definitions & Environmental Operating Prerequisites (Variables) for ManagerView Version 01 Release 5.0

Definition 1 - The Intelligent Workstation Environment (IWSE) Package

1. The IWSE Package is defined as:

   - The Program in machine readable form (the “IWSE Software”)
   - Technical Publications and Technical Communication Material(s) (the IWSE Docuware)

   and subsequent updates as supplied at the sole discretion of Company.

2. The IWSE Package shall additionally be construed to include all copies in machine readable forms including machine output representing the IWSE Software.

Notes

1. ASG cannot and does not represent that its Manager Products will be compatible with any combination of non-ASG products you choose to use them with. While your ASG product supplier may be able to help, you must determine for yourself the compatibility in any particular instance of Manager Products and your hardware/software environment.

2. ASG acknowledges that certain proprietary products or services or programs mentioned within this publication or within other Manager Products publications are distributed under Trademarks or Registered Trademarks of the vendors who own and/or distribute the products or services or programs in the country in question.

Definition 2 - The COREQUISITE Package

1. Full usage of the IWSE Software requires the mandatory presence of the Corequisite ASG-Manager Mainframe Program Products (CMMPP) as Environmental Operating Prerequisites.

2. For the purpose of the IWSE Package the CMMPP is defined as the COREQUISITE Package and is:

   The specified or subsequent Versions of all of these Manager Program Products:

   - ControlManager™ Version 02 Release 4.0
   - DataManager™ Version 01 Release 10.0
   - DictionaryManager™ Version 02 Release 4.0

   operating MVS or OS/390 (all versions) or VM (24 bit mode) or VSE (24 bit mode).

3. These are the associated minimum configurations:

   ControlManager

   - CMR-CM01- ControlManager Nucleus (Version 02)
   - CMR-FE01 - Extended Interactive Facility
   - CMR-WS01- Workstation Interface - Programmable Workstation & Mainframe Tailoring
   - CMR-TP7 - OS/TSO Interface

     and/or

     CMR-TP8 - VM/CMS Interface

     and/or

     CMR-TP2 - CICS Interface

DataManager

DMR-DD1 - DataManager Nucleus (Version 01)

DictionaryManager

- DYR-DY01- DictionaryManager Nucleus (Version 02)
- DYR-TE00- Translation and Transfer Engine
- DYR-TE13- Corporate Dictionary/Repository Definition Export to ManagerView Local Dictionary
- DYR-TI13 - Corporate Dictionary/Repository Definition Import from ManagerView Local Dictionary

The IWSE Order for an ASG-IWSE Package - Attachment 2

3rd Party Environmental Operating Requirements

Use of the PWS/IWSE Package requires that one of the mandatory prerequisite hardware configurations of the following 3rd Party Vendors be met:

Table 27. PWS/IWSE Software Environmental Prerequisite Configurations (see Note 9)

Operating Environment:
Microsoft Windows NT 3.51 or later
Microsoft Windows 95 or later

Table 28. PWS/IWSE Hardware Environmental Prerequisite Configurations

Component	**MSDOS & Windows**
Processor	80,486 or later
Minimum RAM	16Mb
Display	any supported by Windows
Disks	Hard disk (see Note 8) high-density 3.5
Mouse	any supported by Windows
Host Connectivity	see Table 27

Table 29. PWS/IWSE Host Connectivity

	3270 Emulation Product (see Note 7)	Minimum Version
1.	Attachmate EXTRA! for Windows	Version 4.1.0
2.	Banyan Vines Extended SNA Gateway	Version 1.3
3.	Attachmate IRMA WorkStation for Windows	Version 3.0
4.	IBM Personal Communications/3270 for Windows	Version 4.0.0
5.	Novell Netware 3270 LAN Workstation	Version 1.1
6.	Wall Data Rumba for Mainframe	Version 4.0.0

Notes

1. ASG cannot and does not represent that its Manager Products will be compatible with any combination of non-ASG products you choose to use them with. While your ASG product supplier may be able to help, you must determine for yourself the compatibility in any particular instance of Manager Products and your hardware/software environment.

2. Your attention is drawn to Table 27, PWS/IWSE Host Connectivity. The products listed may themselves have certain 3rd Party Host prerequisites. Your ability to utilize your 3270 Emulation Product must be verified with your systems programming department who should be requested to review all the relevant documentation.

3. The tables shown represent ideal Prerequisite Configurations that can be modified by you at your discretion provided you maintain complete compatibility. Your major concern should be to ensure that your installation conforms to the definition of the 3rd Party Software and 3rd Party Hardware that provides the Host Connectivity.

4. Between the date of this publication and its next update ASG may:

   - Introduce support for environments that were not available or were not supported by ASG at the date of publication
   - Withdraw support for environments that were available or were supported by ASG on the date of publication.

   To verify the most up-to-date situation, contact the ASG Service Desk.

5. ASG acknowledges that certain proprietary products or services or programs mentioned within this publication or within other Manager Products publications are distributed under Trademarks or Registered Trademarks of the vendors who own and/or distribute the products or services or programs in the country in question.

6. The information contained herein is subject to change without notice.

7. Table 27 lists certain 3270 Emulation products which provide HLLAPI and/or LLAPI programming interfaces. The ASG PWS/IWSE Packages, MethodManager PWS Graphical Workbench (PGW), and ManagerView (MVW), communicate via these programming interfaces. You will require specific hardware in your PWS/IWSE which is specified by the vendor of the 3270 Emulation product you choose to use. From a PGW/MVW viewpoint the hardware you use to connect to the host may not be significant and it is your decision to use any connection supported by the 3270 Emulation product. Typical connections include co-ax, network using a gateway, synchronous, or asynchronous using modems, etc.

8. A minimum of 5MB of hard disk space is required to install and execute the PWS/IWSE Package; however it is recommended that 10MB of space be available to provide sufficient expansion for application use of the product.

9. Local Area Networks:

   ASG PWS/IWSE products, PGW and MVW, can be run on a Local Area Network as follows:

   - The PWS/IWSE software can be stored on a network drive, providing a consistent version of the products to all users on the LAN.
   - The host connections can be routed through a network gateway.
   - The Local (PWS/IWSE) Dictionary can be stored on a network drive so that any user with access to the LAN (standard security arrangements permitting) can access it. Concurrent access by multiple users will be a supported feature in a subsequent Manager Products release. Currently it is possible for multiple users to have access to the local (PWS/IWSE) Dictionaries, but such use may result in corruption of the content.

Appendix C

Attributes of Datasets Which May Be Supplied on Your Release Tape

Please refer to the table on the following two pages. The abbreviations used are as follows:

Key

DSORG	Dataset Organization
P	Partitioned (dataset)
S	Sequential (dataset)
U	Undefined (record format)
FB	Fixed Blocked (record format)
VB	Variable Blocked (record format)
VBA	Variable Blocked (with ASA control character)

Notes

1. It may be convenient for you to copy all supplied datasets from tape to disk so that the release tape can be freed from use.

2. Dataset MP.CMS must be copied to disk using the CMS TAPPDS command (see"Copying Dataset MP.CMS" on page 27). The space figure given for MP.CMS is a basic requirement only. For an executable version of Manager Products software, additional blocks will need to be allocated. See Chapter 4, "Step 2 - Select and Generate an Executable Version of Manager Products," on page 31 for details.

3. Dataset MP.CMSRCE must be copied to disk using the CMS TAPEMAC command ("Copying Dataset MP.CMSRCE" on page 28).

4. The space figure given for each dataset is in KB and is the necessary free space required in order to copy the dataset to disk.

5. Any sequential files which are to be copied to disk can be copied using the CMS MOVEFILE command.

Attributes of Manager Products Release Datasets that may be Supplied (VM/CMS)

Name	Disk Space Requirement (in KB)	DSORG	Record Format	Record Length	Block Size	When Needed	Tape to Disk Copy Requirement	Contents
MP.CMS	14,400	S	FB	80	23200	Always	Always	CMS text files necessary for installation
MP.CMSRCE	800	P	FB	80	800	When tailoring	Always	Installation macros for tailoring
MP.COM.FIX	160	S	FB	80	8000	To apply fixes to COMMAND and EXECUTIVE members	Recommended	Fixes for application to Manager Products COMMAND and EXECUTIVE-ROUTINE members
MP.COM.UNLOAD	6,000	S	VB	1,024	9,442	Always	Optional	COMMAND, EXECUTIVE-ROUTINE and UDS Table members to be loaded onto the MP-AID
MP.CORP	1,000	S	VB	9,438	9,442	For DB2 and/or SQL, RDG and ADW/IEW Integration facility	Optional	SAVEd source of Corporate Executive Routines used with DB2/SQL Import/Export, Repository Diagram Generation and ADW/IEW Integration facility
MP.DEMO	1,000	S	VB	9,438	9,442	Always	Optional	DEMO Dictionary
MP.DYR.RULES	150	S	VB	9,438	9,442	Corporate Dictionary/ Repository Definition Export for IDD (DYR-TE08)	Optional	SAVEd source of DictionaryManager Translation Rules
MP.FIX	160	S	FB	80	8,000	To apply machine code fixes	Recommended	Fixes for application to Manager Products executable code
MP.INFO	19,000	S	VB	9,438	9,442	For User Defined InfoSystem (CMR-UD10)	Optional	SAVEd copy of InfoDictionary

(continued)

Attributes of Manager Products Release Datasets that may be Supplied (VM/CMS)								
Name	Disk Space Requirement (in KB)	DSORG	Record Format	Record Length	Block Size	When Needed	Tape to Disk Copy Requirement	Contents
MP.INFO.CHANGES	3,000	S	VB	9,438	9,442	For existing User Defined InfoSystem (CMR-UD10) users	Optional	SAVEd copy of InfoDictionary changes
MP.INFO.UNLOAD	13,000	S	VB	1,024	9,442	Always	Optional	InfoBank members to be loaded onto the MP-AID
MP.MMR.ADMIN	17,000	S	VB	9,438	9,442	Always for MethodManager	Optional	SAVEd copy of tailorable MethodManager software and Dictionary/Repository Information Model (DU016) and Administration Repository Information Model (DU777)
MP.MMR.SAMPLE	3,000	S	VB	9,438	9,442	Always for MethodManager	Optional	MethodManager sample repository and SIP Life Cycle
MP.MMR.UNLOAD	24,000	S	VB	1,024	9,442	Always for MethodManager	Optional	Executable MethodManager software
MP.README	140	S	VBA	137	1,370	Always	Optional	READ-ME file for Manager Products latest installation/usage information
MP.SAMPLE.IMPORT	150	S	VB	400	9,442	BACHMAN Generic Import Example	Optional	BACHMAN Generic Import Example
MP.UDS	4,600	S	VB	9,438	9,442	For User Defined Syntax (CMR-UD1)	Optional	SAVEd copy of UDS Table Dictionary

(continued)

Attributes of Manager Products Release Datasets that may be Supplied (VM/CMS)								
Name	Disk Space Requirement (in KB)	DSORG	Record Format	Record Length	Block Size	When Needed	Tape to Disk Copy Requirement	Contents
MP.UIBAL.CMS	3,500	P	FB	80	23,200	For User Interface (CMR-UI1)	Always	Assembler data descriptions that can be copied into the user program
MP.UICOB.CMS	3,800	P	FB	80	23,200	For User Interface (CMR-UI1)	Always	COBOL data descriptions that can be copied into the user program
MP.UIDICT	5,000	S	VB	9,438	9,442	For User Defined Output (DYR-UD15) and User Interface (CMR-UI1)	Optional	SAVEd copy of User Interface Dictionary
MP.UIMIV.CMS	2,200	P	FB	80	23,200	For User Interface (CMR-UI1)	Always	MARK IV data descriptions that can be copied into the user program
MP.UIPLI.CMS	4,200	P	FB	80	23,200	For User Interface (CMR-UI1)	Always	PL/I data descriptions that can be copied into the user program

Appendix D

Attributes of ASG-supplied Dictionaries

This appendix provides information about those dictionaries that can be generated from the sequential SAVEd datasets which are supplied on the release tape.

The space allocation figure for each dictionary dataset indicates the number of physical disk blocks required to RESTORE the dictionary. It includes a small space allowance to enable the user to expand the dictionary if required. If you wish to make large additions to the dictionary, you should increase the space allocations accordingly.

The number of physical blocks required for each dictionary dataset have been calculated using these logical and physical blocksizes:

Dataset	**Logical blocksize**	**Physical blocksize**
Index	2,046	8,192
Data Entries	360	8,192
Source	314	4,096
Recovery	-	4,096
Log	-	4,096

These blocksizes can be used for all ASG-supplied dictionaries in any supported environment and provide reasonable utilization of DASD space. For a specific environment or disk type the blocksizes and space allocations can be adjusted to suit the user's particular requirements.

Refer to Chapter 7, "Step 5 - Set Up Dictionaries," on page 81 for further details on installing these dictionaries.

Attributes of ASG-Supplied Dictionaries							
Attribute	Administrative Dictionary (Note 3 & Note 4)					DEMO Dictionary	User Interface Dictionary
File Name	MP.UDS (Note 3)	MP.CORP	MP.INFO	MP.INFO.CHANGES	MP.DYR.RULES	MP.DEMO	MP.UIDICT
Number of Statuses	6 (Note 5)	-	-	-	-	14 (Note 5)	6 (Note 5)
Index Size	20	20	50	25	5	20	30
Data Entries Size	500	100	1200	250	20	220	950
Source Size	800	400	4700	750	30	270	750
Recovery Size	100	20	120	100	10	120	20
Log Size (Note 1)	50	50	50	50	50	50	50
Keyword used on RESTORE	ALL (Note 2)	SOURCE	SOURCE	SOURCE	SOURCE	ALL (Note 2)	ALL (Note 2)

Notes

1. The space allocation for the Log file is determined by the type of logging (updates only, or all commands) and frequency of use, and is not a factor of dictionary size. ASG recommends a minimum value of 50 blocks of 4,096 bytes each to avoid operational problems which can arise with full log situations.

2. If the keyword used on RESTORE is ALL the file must be RESTOREd to an empty dictionary.

3. If you restore the Administration dictionary, file MP.UDS must be loaded first.

4. For the Administrative dictionary, the number of blocks required for the Index, Data Entries, Source, and Recovery files will vary according to which and how many ASG-supplied dictionaries/files you RESTORE.

5. This number represents the minimum statuses that must be defined at dictionary creation time in order to RESTORE ALL successfully from the input SAVE ALL file. You must allow for any additional user statuses required and increase the value accordingly.

Attributes of ASG-Supplied Repositories (for MethodManager)		
Attribute	MethodManager ADMINISTRATION Repository	MethodManager SAMPLE Repository
Dataset Name	MP.MMR.ADMIN	MP.MMR.SAMPLE
Number of Statuses	4 (Note 3)	5 (Note 3)
Index Size	50	15
Data Entries Size	1000	300
Source Size	2500	500
Recovery Size	50	50
Log Size (Note 1)	50	50
Keyword used on RESTORE	ALL (Note 2)	ALL (Note 2)

Notes

1. The space allocation for the Log dataset is determined by the type of logging (updates only, or all commands) and frequency of use, and is not a factor of repository size. We recommend a minimum value of 50 blocks of 4,096 bytes each to avoid operational problems which can arise with full log situations.

2. If the keyword used on RESTORE is ALL the dataset must be RESTOREd to an empty dictionary.

3. This number represents the minimum statuses that must be defined at repository creation time in order to RESTORE ALL successfully from the input SAVE ALL dataset. You must allow for any additional user statuses required and increase the value accordingly.

Appendix E

Virtual Storage Requirements for Running Manager Products

Introduction

The minimum virtual machine size required to run any version of your Manager Products can be ascertained by adding together:

- The virtual storage requirements for running the Manager Products program code (see "Storage Requirements for Running Manager Products Program Code" on page 157)
- The requirement for dynamically allocated virtual storage ("Storage Requirements for Dynamically Allocated Virtual Storage" on page 159)
- The virtual storage requirement for input/output buffers ("Storage Requirements for Input/Output Buffers" on page 160)
- The requirement for running Access Call programs (see "Additional Storage Requirements" on page 161)
- The virtual storage required by CMS. 300 to 400K would be appropriate.

Storage Requirements for Running Manager Products Program Code

The storage needed is dependent upon:

- Which installation option is selected
- Which Manager Products you have purchased
- The degree of Manager Products tailoring
- The commands to be used.

Virtual storage requirements for the Manager Products program code are given in this table:

Manager Products Program Code Virtual Storage Requirements					
Manager Products Components	**Primary Load Module Names**	**Execution Option 1 (Kilobytes)**	**Execution Option 2 (Kilobytes)**	**Execution Option 2 (Kilobytes)**	**Remarks**
ControlManager/ DataManager/ DictionaryManager	MPR00	600-2,000 (Note 1)	300	2,000	
MethodManager	MMWKB	1,000	1,000	1,000	
	MMRSY				
	MGA01				
	MGA02				
	MGA03				
If DesignManager commands are to be issued, the following additional storage is required					
DesignManager (non-overlay)	DSR00	1,280	1,280	1,280	See Note 1 & Note 2
DesignManager (overlay)	DSR00	760	760	760	See Note 1 & Note 2

Notes

1. The actual program storage required is dependent upon the Manager Products functionality used and the residency attribute of the code segments loaded.

2. The DesignManager code will not be loaded until the first DesignManager command is issued.

3. The storage value given is for DesignManager as supplied by ASG. It assumes that no tailoring of buffer sizes has taken place using the LBUF1 installation macro. The default values for DesignManager are given in this table:

Default DesignManager Virtual Storage Requirements		
Storage Area	Size (in kilobytes)	
	Non-overlay	Overlay
DesignManager program storage excluding buffers	1,000	480
Workbench design area (WBSIZE)	250	250
Format area (FMSIZE)	10	10
Member type table (MTTMAX)	20	20
Total virtual storage requirement	1,280	760

Note:

WBSIZE, FMSIZE, and MTTMAX are the LBUF1 keywords used to tailor the workbench design area, the format area and the member type table respectively. If these buffers are tailored using LBUF1, the DesignManager storage requirement must be adjusted accordingly. See "DesignManager Installation Macros" on page 66 for details of the LBUF1 installation macro.

Storage Requirements for Dynamically Allocated Virtual Storage

An allowance of 200 to 300K should be made to cover Manager Products requirements for dynamically allocated virtual storage. This will be sufficient in most cases, although certain commands (for example REPORT, HIERARCHY, and PRODUCE) that operate along reference paths with a large number of members can exceed this figure, as can the generation of extensive KEPT-DATA lists. In full-screen interactive environments when a high number of output lines are generated, the requirements for dynamically allocated storage can increase significantly.

The QUERY VIRTUAL command can be used at any time during a ControlManager run to ascertain the maximum amount of dynamically allocated storage that has been used and the amount that is currently retained.

Storage Requirements for Input/Output Buffers

A buffer or buffer pool is allocated for each Manager Products dataset to be used. The basic requirements are these:

- At least two buffers for each of the dictionary's Source, Index, and Data Entries buffer pools
- At least two buffers for the primary MP-AID buffer pool
- At least two buffers for each secondary MP-AID buffer pool
- One buffer for the dictionary's Recovery dataset
- If logging is active, one buffer for the dictionary's Log dataset.

Specification of additional buffers can lead to a considerable reduction in dataset input/output activity, resulting in greatly improved command response times. The use of extended virtual storage allows much larger buffer pools to be allocated (up to 32,000 if there is sufficient virtual storage available). This can lead to further improvements in performance and response times.

For full details of the effects of specifying additional buffers, refer to *ASG-Manager Products Performance Tuning*.

The amount of virtual storage required for each buffer is equal to the physical blocksize of the relevant dataset. The blocksize for the MP-AID is determined by the Systems Administrator when the MP-AID is created (see Chapter 5, "Step 3 - Tailoring Installation Macros and Modules Supplied in MP.CMSRCE," on page 37). Blocksizes for the dictionary datasets are defined by the Controller when the dictionary is created (see Chapter 6, "Step 4 - Create and Load the MP-AID," on page 75). The size of a given buffer pool is thus derived by multiplying the dataset physical blocksize by the number of buffers to be allocated to the buffer pool.

If two or more dictionaries are to be used in a job, only one will be open at any one time.

The buffer size for each further dataset declared in a FILEDEF statement in the job control statements must also be added: that is, the buffer size of dataset MPIN (for the primary input device) and of dataset MPOUT (for the primary output device). Total virtual storage requirements for input/output buffers may be ascertained by adding together the buffer or buffer pool sizes of the following datasets:

- For the dictionary:
  - Index dataset
  - Source dataset
  - Data Entries dataset
  - Recovery dataset
  - Log dataset
- The primary MP-AID dataset
- Each secondary MP-AID dataset
- The MPIN dataset
- The MPOUT dataset.

Additional Storage Requirements

For Input/Output Buffers

Particular Manager Products (for example, DesignManager) and certain optional additional facilities (for example, the User Interface facility) require additional input or output datasets. The buffer sizes of these additional datasets must be added to the virtual storage requirements when these products/facilities are being used. The usage of these additional datasets can be found in Chapter 9, "Step 7 - Linking External Software To Manager Products Software," on page 95 and Chapter 10, "Step 8 - Run Manager Products with BDAM MP-AID and BDAM Dictionaries," on page 101. Double buffering is used for all input and output sequential datasets. Default blocksizes for these datasets can be found in Appendix F,"Virtual Storage Requirements for Running Manager Products," on page 157 and Appendix G,"Attributes of Manager Products Output Datasets," on page 165.

Some Controller's and Systems Administrator's commands also require additional datasets (see Chapter 9, "Step 7 - Linking External Software To Manager Products Software," on page 95 and Chapter 10, "Step 8 - Run Manager Products with BDAM MP-AID and BDAM Dictionaries," on page 101); if any of these commands are to be used during a run, then an allowance must be added for the buffers of the datasets they use.

When Running Access Call Programs

When running Access Call programs, additional storage will be required for:

- The user program storage requirement
- The ASG-supplied interface module (DMRUS)—approximately 2K.
- The Access Call buffer and work file buffers, if used. (See the appropriate User Interface Facility publication.)

Appendix F

Attributes of Manager Products Input Datasets

Attributes of Manager Products Input Datasets				
Purpose	**Dataset Organization**	**Record Format**	**Record Length**	**Blocksize**
For Automation of Set Up (DMR-AS1, DMR-AS2)	Partitioned	FB	80	Any
For RESTORE commands	Sequential	VB	9,438	Any
For RELOAD commands	Sequential	VB	Any (Note 1)	Any
For RELOAD or ROLL-FORWARD commands	Sequential	VB	9,438	Any
For LOG ARCHIVE commands (optional)	Sequential	VB	9,438	Any
For LOG ANALYSIS or AUDIT commands (optional)	Sequential	VB	9,438	Any
For MP-AID LOAD or MP-AID RELOAD commands	Sequential	VB	Any (Note 1)	Any
For Generic Import	Sequential Partitioned	FB/VB FB/VB	Any Any	Any Any

Notes

1. The record length is determined when the dataset is created.

Appendix G

Attributes of Manager Products Output Datasets

Attributes of Manager Products Output Datasets				
Purpose	**Dataset Organization**	**Record Format**	**Record Length**	**Default Blocksize**
For DictionaryManager TRANSFER command	Sequential	FB	80	400 (Note 1)
For Source Language Generation (DMR-SL1 to DMR-SL9) facility	Sequential	FB	80	400 (Note 1)
	Partitioned	FB/VB	(Note 4)	(Note 4)
For User Interface (CMR-UI1)	Sequential	VB	700	9,442 (Note 1 & Note 2)
For SAVE Command	Sequential	VB	9,438	9,442 (Note 1 & Note 2)
For UNLOAD Command	Sequential	VB	(Note 3)	9,442 (Note 1 & Note 2)
For LOG ARCHIVE Command	Sequential	VB	284	9,442 (Note 1 & note 2)
For AUDIT ONTO Command	Sequential	VB	255	9,442 (Note 1 & Note 2)
For MP-AID UNLOAD Command	Sequential	VB	MP-AID blocksize	9,442 (Note 1 & Note 2)
For SENDF Executive command	Sequential	FB/VB	Min 1 Max 32,760	Min 1 Max 34,760
	Partitioned	FB/VB	(Note 4)	(Note 4)

Attributes of Manager Products Output Datasets				
For REPLAY facility MPRECORD	Sequential	VB	4,096	9,442 (Note 1 & Note 2)
MPRESULT	Sequential	VB	4,096	9,442 (Note 1 & Note 2)
MPSPRINT	Sequential	VBA	137	9,442 (Note 1 & Note 2)
For TRACE output	Sequential	VB	137	9,442 (Note 1 & Note 2)

Notes

1. The BLOCK or BLKSIZE parameter can be used in order to vary the default blocksize if required.

2. Any specified blocksize must be at least 4 bytes greater than the Record Length. The Record Length will be the largest blocksize of the Index, Source, and Data Entries datasets for the relevant dictionary.

3. The Record Length will be the largest blocksize of the Index, Source, and Data Entries datasets for the relevant dictionary.

4. The Record Length and Blocksize are determined when the partitioned dataset is created.

Appendix H

Applying Fixes to Manager Products Software

Introduction

The purpose of this appendix is to provide you with the information necessary for the correct and successful application of software fixes which have been made available using the following two sources:

- The appropriate migration notice
- Your local Manager Products supplier.

ASG may also supply you with fixes using these three sources, where in each case the appropriate fix installation information is included in that source:

- In dataset MP.FIX (which may be supplied on the Release tape),
- In dataset MP.COM.FIX (which may be supplied on the Release tape),
- On a floppy disk.

The fixes to be applied fall into two categories:

- Manager Products fixes
- Product-specific fixes.

Fixes can be applied to Manager Products:

- Machine code, using the CMS ZAP command
- COMMAND members, using the MZAP command
- Dictionary source of MP-AID EXECUTIVE members, using the MODIFY command.

Manager Products Fixes

Manager Products fixes should be applied by all users, irrespective of products and selectable units installed. The fix identifier for Manager Products fix items takes the form:

`CMR/V`*`vv`*`/REL`*`rrr`*`/`*`nnn`*

where

vv is the version number of the Manager Products.

rrr is the Release number of the Manager Products.

nnn is the fix number within the Release.

The fix may have to be applied to the ControlManager program or the DataManager program.

Product-Specific Fixes

When a fix item from a Software Notice is not designated as Manager Products, any associated fix need only be applied by users of the specific Product or Product selectable unit. The fix item will identify the relevant area. The fix identifier takes the form:

ppp/Vvv/REL*rrr*/*nnn*

where

ppp is the Manager Products name:

DYR for DictionaryManager

DMR for DataManager

DSR for DesignManager

MMR for MethodManager

vv is the version number of the Manager Products.

rrr is the Release number of the Manager Products.

nnn is the fix number within the Release.

Applying Fixes

Fixes supplied by your local Manager Products supplier may be provided:

- As Procedures Language updates, for application to COMMAND members on the MP-AID and/or the dictionary source of MP-AID EXECUTIVE members,
- As machine code fixes for application to the Manager Products program code.

Fixes provided by ASG for application to COMMAND members should be applied using the MZAP command, details of which are supplied in "Applying Fixes to ASG-Supplied COMMAND Members Using the MZAP Command" on page 169.

Fixes provided by ASG for application to the source version of EXECUTIVE-ROUTINE members (supplied in dataset MP.CORP or MP.MMR.ADMIN) should be applied using the MODIFY command, details of which are supplied in "Applying Fixes to the ASG-Supplied EXECUTIVE Members Using the MODIFY Command" on page 174.

ASG-supplied machine code fixes are normally in a form suitable for processing by the CMS ZAP command.

The application of a fix is divided into two areas:

- The application of the actual fix
- The recording of that application.

A display of recorded fixes can be produced using the ENVIRONMENT command. The output from the ENVIRONMENT command will be required by ASG if you should ever encounter a problem with Manager Products software. In some cases, recording applied fixes can be as important as applying the fix itself, particularly in a problem solving situation.

You should register the application of a fix in a central recording table called MPFID. It is important that fixes are only registered after they have been applied to all installed versions of your Manager Products software to avoid the possibility of receiving misleading output from the ENVIRONMENT command.

Applying Fixes to ASG-Supplied COMMAND Members Using the MZAP Command

Introduction

ASG supplies some software functionality in the form of COMMAND members in datasets MP.COM.UNLOAD and MP.MMR.UNLOAD. These COMMAND members are intended to be loaded onto the MP-AID at installation time. From time to time, ASG may supply you with details of fixes to be applied to these COMMAND members. ASG supplies a special Systems Administrator command, called MZAP, to facilitate the application of these fixes.

There are six variants of the MZAP command. Four of these variants must be issued for each COMMAND member to be fixed in the sequence given below.

Step 1 . Issue the MZAP XSELECT variant.

Step 2. For each line to be modified within a COMMAND member, issue the MZAP VERIFY variant followed by the MZAP REPLACE variant.

Step 3. Issue the MZAP APPLY variant.

The two remaining variants are only required in occasional circumstances, generally in error situations. These variants are as follows:

MZAP HISTORY

MZAP CANCEL

The clauses used with each variant are described in more detail below followed by details of messages which may be encountered.

Step 1 - Issue the XSELECT Clause

Syntax:

```
MZAP XSELECT member-name;
```

where *member-name* is the name of the MP-AID COMMAND member to be fixed.

This clause checks for the existence of the specified COMMAND member on the MP-AID. If it exists, it:

- Brings a copy into virtual storage
- Issues message number MZ0000.

Note that if the selected member is not found on the primary MP-AID then, if the member is present on a secondary concatenated MP-AID, the member is copied from the secondary to the primary MP-AID.

Secondary MP-AIDs are searched in the order specified by the MP-AID CONCATENATION command. The member is copied from the first occurrence found. Message MZ0000 is replaced by message MZ0013.

Step 2A - Issue the VERIFY Clause

Syntax:

```
MZAP VERIFY line-number hexadecimal-verify-strings;
```

This clause attempts to match the specified strings against the contents of the line (identified by the specified line-number) in the virtual storage copy of the COMMAND member.

This is intended to ensure that any subsequent modification using the MZAP REPLACE variant is made to the correct line. Up to 8 strings may be specified in any MZAP VERIFY statement. The specified strings are concatenated to form the verification text for the specified line.

If the verification is successful, then message number MZ0014 is issued.

The MZAP VERIFY and MZAP REPLACE variants must be issued for each line in the COMMAND member which is to be modified.

Step 2B - Issue the REPLACE Clause

Syntax:

```
MZAP REPLACE line-number hexadecimal-replace-strings;
```

This clause replaces the contents of the line (identified by the specified line-number) with the specified strings. MZAP REPLACE can only be issued after successfully issuing the MZAP VERIFY variant. Up to 8 strings may be specified in any MZAP REPLACE statement. The specified strings are concatenated to form the replacement text for the specified line.

If the replacement is successful, then message number MZ0019 is issued.

The MZAP VERIFY and MZAP REPLACE variants must be issued for each line in the COMMAND member which is to be modified.

Step 3 - Issue the APPLY Clause

Syntax:

```
MZAP APPLY checksum;
```

This clause replaces the original version of the COMMAND member on the primary MP-AID with the modified version set up in virtual storage and terminates processing of the selected COMMAND member.

An 8- or 11-digit hexadecimal *checksum* must be entered and ensures that the previously entered VERIFY and REPLACE strings have been entered correctly.

In addition to replacing the updated COMMAND member on the primary MP-AID the COMMAND member £PCMMZHIST is updated with details relating to the current update and thus provides an audit trail of updates applied to all COMMAND members on the MP-AID. The MZAP HISTORY command is used to display the contents of this member.

If the application is successful, then message number MZ0001 is issued.

The CANCEL Clause

Syntax:

```
MZAP CANCEL;
```

This clause allows you to cancel MZAP processing at any stage between entering the MZAP XSELECT variant and the MZAP APPLY variant. It will cause the virtual storage copy to be erased. If the member to be updated was copied from a secondary concatenated MP-AID then the copied member is also deleted from the primary MP-AID.

The HISTORY Clause

Syntax:

```
MZAP HISTORY;
```

This clause provides an audit trail of fixes applied to all COMMAND members present on the primary MP-AID.

For each successful MZAP executed an entry is made consisting of the COMMAND member name, fix date and time, Manager Products fix number and checksum information.

Note that entries for fixes issued on a provisional basis, i.e. without an authorized ASG fix number, contain the string *???* in place of the fix number.

The MZAP HISTORY variant will generally only be needed in the event of failure of the MZAP VERIFY variant to successfully verify a particular line in the virtual storage copy of a COMMAND member. The MZAP HISTORY variant can be used to establish whether the correct number of previous modifications have been applied to that COMMAND member.

MZAP Output Messages

A number of messages may be output when using the MZAP command.

`MZ000 MEMBER member-name HAS BEEN SELECTED`
: This message indicates that the COMMAND member specified with the MZAP XSELECT variant exists and has been copied into virtual storage. You should now issue the MZAP VERIFY and MZAP REPLACE variants for each line to be modified.

`MZ0001 MEMBER member-name updated, MZAP PROCESSING ENDED`
: This message is issued if the MZAP APPLY variant has been successfully processed, that is, the original MP-AID copy of the COMMAND member has been overwritten by the updated virtual storage copy. The history COMMAND member £PCMMZHIST has also been updated.

`MZ0002 MZAP PROCESSING TERMINATED`
: This message is issued as the result of the MZAP CANCEL variant to confirm that MZAP processing for a particular COMMAND member has terminated.

`MZ0003 NO VALID KEYWORD FOUND`
: This message indicates that you have issued the MZAP command without a valid keyword.

`MZ0004 ERROR CODE message-number RETURNED FROM MZAP`
: This message indicates that a standard Manager Products message identified by the specified message number was generated by the MZAP command. You should refer to the *ASG-Manager Products Message Guide*, or, alternatively, issue a PANEL command followed by the indicated message number to access the appropriate InfoBank panel for further details of this particular message.

`MZ0005 MEMBER NAME OMITTED`
: This message indicates that you have not supplied the name of a COMMAND member when required with a particular MZAP command.

`MZ0007 NO CHANGES MADE, APPLY NOT ACTIONED, MZAP PROCESSING ENDED`
: This message indicates that no lines have yet been changed in the copy of the COMMAND member in virtual storage, so that there is no purpose in the MZAP APPLY variant just issued being executed.

`MZ0008 INVALID LINE NUMBER SPECIFICATION`
This message indicates that a non-numeric line number was specified with either the MZAP VERIFY or the MZAP REPLACE variants.

`MZ0009 LINE NUMBER OUTSIDE RANGE`
This message indicates that a line number was specified with either the MZAP VERIFY or MZAP REPLACE variants, which was greater than the total number of lines in the COMMAND member.

`MZ0010 NO MEMBER CURRENTLY SELECTED`
This message indicates that any of these MZAP keywords has erroneously been entered without having previously selected a COMMAND member using the MZAP XSELECT variant:

VERIFY

REPLACE

APPLY

CANCEL

`MZ0011 INVALID KEYWORD keyword COMMAND IGNORED`
This message indicates that you have entered the specified invalid keyword with the MZAP command.

`MZ0012 TOO MANY VERIFY/REPLACE STRINGS SPECIFIED`
The number of strings specified in the MZAP VERIFY/REPLACE variant exceeds the maximum number of 8.

`MZ0013 MEMBER member-name HAS BEEN COPIED AND SELECTED`
This message indicates that the COMMAND member specified with the MZAP XSELECT variant has been copied from a secondary to the primary MP-AID and made available in virtual storage. The MZAP VERIFY and REPLACE variants can now be issued.

`MZ0014 VERIFY ACCEPTED`
This message indicates that the MZAP VERIFY variant has successfully verified the contents of the specified line in the virtual storage copy of the selected COMMAND member.

`MZ0015 VERIFY STRING NOT FOUND AT LINE line-number`
`MZ0016 VERIFY DATA FOLLOWS`
hexadecimal verify data
`MZ0017 ACTUAL DATA FOLLOWS`
hexadecimal member data
These three messages indicate that the MZAP VERIFY variant did not process successfully because the verification string did not match the current contents of the line specified with the MZAP VERIFY variant.

`MZ0018 LINE NOT PREVIOUSLY VERIFIED`
: This message indicates that an attempt has been made to change a line in the virtual storage copy of a selected COMMAND member using the MZAP REPLACE variant, where that line has not been previously verified using the MZAP VERIFY variant.

`MZ0019 REPLACE ACCEPTED`
: This message indicates that the MZAP REPLACE variant has been successfully processed and that the specified change has been made to the copy of the selected COMMAND member in virtual storage.

`MZ0020 CHECKSUM OMITTED OR INVALID`
: This message indicates that the mandatory checksum specified with the MZAP APPLY variant has either been omitted or is incorrect for the previously entered MZAP VERIFY and REPLACE variants. The correct checksum must be specified or any incorrectly specified MZAP VERIFY/REPLACE statements must be corrected and the MZAP rerun.

`MZ0024 MPAID IN READ-ONLY MODE, MZAP PROCESSING ENDED`
: The MP-AID is currently open in read-only mode. The MZAP command requires access to a read/write MP-AID.

Applying Fixes to the ASG-Supplied EXECUTIVE Members Using the MODIFY Command

ASG supplies some functionality in the form of EXECUTIVE members in datasets MP.COM.UNLOAD and MP.MMR.UNLOAD. These EXECUTIVE members are intended to be loaded onto the MP-AID at installation time. Source versions of these members are supplied as dictionary EXECUTIVE-ROUTINE members in datasets MP.CORP and MP.MMR..ADMIN. This allows user modification of MP-AID EXECUTIVE members by:

- Modifying the dictionary EXECUTIVE-ROUTINE members and
- Constructing the modified version onto the MP-AID to replace the equivalent EXECUTIVE member on the MP-AID.

From time to time, we may supply details of fixes to be applied to the dictionary EXECUTIVE-ROUTINE members for subsequent constructing.

The supplied fixes take no account of any user modifications that may have been made to an EXECUTIVE-ROUTINE member. Thus, when applying a fix to a particular EXECUTIVE-ROUTINE member, it is important that you make allowances for any user modifications that have been implemented.

Applying Machine Code Fixes in a CMS Environment

Machine code fixes are applied in CMS Environments using the CMS ZAP command. Fixes supplied by your local Manager Products supplier are normally in a form suitable for processing by the CMS ZAP command. The CMS ZAP command may be used interactively or run from a CMS file by using the INPUT option of the ZAP command. The CMS file must have a file-type of ZAP.

A fix is first applied, then the fix application is recorded. Application of the fix is dependent on the installation option you select. The table following shows the three execution options available, the format of the ZAP command required and the applicable program names.

A display of recorded fixes can be produced using the ENVIRONMENT command. The output from the ENVIRONMENT command will be required by ASG if you should ever encounter a problem with Manager Products software. In some cases, recording of applied fixes can be as important as the application of the fix itself, particularly in a problem solving situation.

You should register the application of a fix in a central recording table called MPFID. It is important that fixes are only registered after they have been applied to all installed versions of your Manager Products software to avoid the possibility of receiving misleading output from the ENVIRONMENT command.

Manager Products Component		**1 LOADLIB Mode**	**2 Dcss Mode (Note 2)**	**3 GENMOD Mode**
CMR/DMR/DYR	ZAP Command	ZAP LOADLIB MPRLIB	ZAP MODULE	ZAP MODULE
	Applicable Program Names	MPR00-80	MPR00 (Note 3)	MPR00 (Note 3)
DSR	ZAP Command	ZAP LOADLIB MPRLIB	ZAP LOADLIB MPRLIB	ZAP LOADLIB MPRLIB
	Applicable Program Names	DSR00	DSR00	DSR00
MMR	ZAP Command	ZAP LOADLIB MPRLIB	ZAP LOADLIB MPRLIB	ZAP LOADLIB MPRLIB

Manager Products Component		1 LOADLIB Mode	2 Dcss Mode (Note 2)	3 GENMOD Mode
MMR	Applicable Program Names	MMxxx	MMxxx	MMxxx
Dynamically Loaded Text File	ZAP Command	ZAP TEXTLIB txtlib-name (Note 1)	ZAP TEXTLIB txtlib-name (Note 1)	ZAP TEXTLIB txtlib-name (Note 1)
	Applicable Text file names	text-file-name	text-file-name	text-file-name

Notes

1. Tables, maps, etc. which are dynamically loaded by a Manager Products exist as text files on the mini-disk containing the Manager Products software. A fix cannot be applied directly to these files. You must first create a CMS TXTLIB containing the text file to be fixed. The fix may then be applied and the text file re-created from the TXTLIB member. Dynamically loaded text files can be determined by the annotation '(DYNAMICALLY LOADED)' on the NAME statement for the fix.

2. If you select execution option 2, after applying the fix or fixes to MPR00, the shared segment must be reloaded by using a LOADM MPR00 command and a CP SAVESYS/SAVESEG CMSS2210 command as described in "Generation of the Option 2 Executable Version" on page 33. Note that the generation of the module MPR00 using the CMS exec MPX2 must NOT be performed when applying fixes. This exec is only used at installation time.

3. For execution options 2 and 3, references within a fix to MPR00-MPR80 must be replaced by MPR00.

Index

P

R

S

U

Z